Report No. DOT/FAA/RD-82/86

(12)

Project Report
ATC-115

AD A139145

General Aviation TCAS Avionics (GATCAS)

R.L. Briggs
J. DiBartolo
D.L. Underwood
D.J. Callahan

17 February 1984

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS

Prepared for the Federal Aviation Administration

Document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

Best
Available
Copy

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT-FAA-RD-82/86	AD-A137 145	

4. Title and Subtitle
General Aviation TCAS Avionics (GATCAS)

5. Report Date
17 February 1984

6. Performing Organization Code

7. Author(s)
Randal L. Briggs
Joseph DiBartolo
Donald I. Underwood
Dennis J. Callahan

8. Performing Organization Report No.
ATC-115

9. Performing Organization Name and Address
Lincoln Laboratory, M.I.T.
P.O. Box 73
Lexington, MA 02173-0073

10. Work Unit No.

11. Contract or Grant No.
DOT-FA72WAI-877

12. Sponsoring Agency Name and Address
Department of Transportation
Federal Aviation Administration
Systems Research and Development Service
Washington, DC 20591

13. Type of Report and Period Covered
Project Report

14. Sponsoring Agency Code

15. Supplementary Notes

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology, under Air Force Contract F19628-80-C-0002.

16. Abstract

Experimental Traffic Alert and Collision Avoidance System (TCAS) avionics developed for the FAA at M.I.T. Lincoln Laboratory are described. The objective of the program under which this equipment was developed was to assess the feasibility of providing a small, low-cost unit for general aviation usage.

The experimental general aviation TCAS (GATCAS) avionics incorporates a new system architecture using a microprogrammed sequencer, a 16-bit microprocessor and a low-power, solid-state transmitter appropriate to the class of aircraft expected to employ GATCAS.

The general aviation unit is designed to operate below 10,000 feet in densities of up to 0.02 aircraft/nmi^2, and to provide a pilot warning time (TAU$_R$) of 25 seconds. Assuming a track acquisition time of 10 seconds and a maximum closing speed of 300 knots, the required theoretical range of GATCAS is 3.4 nmi.

The report includes (as an appendix) a cost analysis for general aviation TCAS avionics.

17. Key Words
ATC avionics
FAA beacon equipment
Threat alert
Collision avoidance
Equipment cost

18. Distribution Statement
Document is available to the public through the National Technical Information Service, Springfield, Virgina 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	104	

Form DOT F 1700.7 (8-69)

CONTENTS

	Page
1.0 INTRODUCTION	1
2.0 GATCAS DESIGN	2
2.1 Overview	2
2.1.1 Design Philosophy	2
2.1.2 System Operating Characteristics	3
2.1.3 Major System Elements	3
2.2 RF Unit	7
2.2.1 Microwave Signal Source (1030 MHz)	7
2.2.2 PAM/DPSK Modulators	9
2.2.3 Microwave Power Module	9
2.2.4 Other Transmitter RF Components	9
2.2.5 Receiver	11
2.3 Video-Pulse Quantizer	11
2.4 Reply Detector	14
2.4.1 General Description	14
2.4.2 Digitizer Clock Control	18
2.4.3 Video Digitizer	18
2.4.4 Sliding Window Detector	18
2.4.5 Mode S Preamble Detector	25
2.4.6 Pseudo Leading Edge Generation and Short Pulse Rejection	25
2.5 Reply and Interrogation Controller	31
2.5.1 General Description	31
2.5.2 Am 2910 Microprogram Sequencer	31
2.5.3 Am 2914 Interrupt Controller	34
2.5.4 Microprogram Memory	35
2.5.5 Control Register	35
2.5.6 Status Register	38
2.5.7 Range Counter Latch	38
2.5.8 Mode S Data Memory	38
2.6 Mode S Interrogation Generator (DIG)	41
2.6.1 General Description	41
2.6.2 Mode S DPSK Encoder	41
2.7 Mode S Reply Processor (DRP)	41
2.7.1 General Description	41
2.7.2 Mode S Downlink Decoder	43
2.8 Mode C Reply Accumulator	43
2.8.1 General Description	43
2.8.2 CRA Storage Memory	43
2.9 Computer Subsystem	46
2.9.1 General Description	46
2.9.2 Z8002 Microprocessor	46
2.9.3 System Memory	48
2.9.4 I/O Address Space	48
2.9.5 Am8255A Parallel I/O Port	48

CONTENTS (Cont'd)

	Page
2.9.6 Am9551 Serial I/O Port	52
2.9.7 Am9519A Interrupt Controller	52
2.9.8 Am9513 System Timing Controller	52
2.9.9 Am8253 Counter Timer	52
2.9.10 Bus Connection Logic	53
3.0 GATCAS SOFTWARE	54
3.1 Overview	54
3.2 Real-Time Microprocessor Software	54
3.2.1 Task Scheduler and I/O Handler	54
3.2.2 Task Directives	55
3.2.2.1 Create Task (RTSK)	55
3.2.2.2 Mark Wait (MARKWT)	55
3.2.2.3 Mark Return (MARKRT)	55
3.2.2.4 Update Time (UPDTM)	55
3.2.2.5 Wait for Flag (WAITF)	56
3.2.2.6 Change Priority (CBGPRI)	56
3.2.2.7 Transmit (XMITR)	56
3.2.2.8 Receiver (RCVR)	56
3.2.2.9 Flush Buffer (FLUSH)	57
3.2.2.10 Simulate Input (SIMIN)	57
3.2.3 Mode C Processing	57
3.2.4 Mode S Processing	60
3.2.5 Squitter Processing	60
3.2.6 TEU Data Communications	60
3.2.6.1 Interface Protocol and Formats	60
3.2.6.2 TEU Input and Output Handlers	61
3.2.6.3 GATCAS Input and Output Tasks	61
3.3 Real-Time Microprogram Software	64
3.3.1 General Description	64
3.3.2 Mode C Processing	64
3.3.3 Mode S Processing	64
3.3.4 Squitter Processing	70
4.0 DIAGNOSTIC HARDWARE	72
5.0 DIAGNOSTIC SOFTWARE	74
5.1 General Description	74
5.2 Z8002 System Diagnostics	74
5.3 RIC Diagnostics	75
5.3.1 Mode C Diagnostics	76
5.3.2 Mode S Diagnostics	77
5.4 TEU Diagnostics	77

CONTENTS (Cont'd)

	Page
6.0 FLIGHT TEST RESULTS	79
6.1 Mode S Performance	79
6.2 Mode C Mode Performance	79
6.2.1 Detection at Long Range	79
6.2.2 Statistical Performance Assessment	83
6.2.2.1 Performance Definitions	83
6.2.2.2 Probability of Report	83
6.2.3 Probability of Track	85
6.2.4 Performance as a Function of Aircraft Density	85
REFERENCES	92
APPENDIX A GATCAS COST ESTIMATE	A-1

ILLUSTRATIONS

Fig.		Page
2.1-1	GATCAS collision avoidance unit	4
2.1-2	GATCAS operational configuration	5
2.1-3	GATCAS architecture	6
2.2-1	General aviation TCAS microwave block diagram	8
2.2-2	Typical DPSK phase transition (180° phase reversal). Input to 150W amplifier (a), output of 150 watt amplifier (b).	10
2.2-3	GATCAS receiver log video response	13
2.3-1	VPQ block diagram	15
2.3-2	VPQ waveforms	16
2.3-3	ATCRBS DMTL	17
2.3-4	Mode S DMTL	17
2.4-1	Digitizer clock control	19
2.4-2	Digitizer logic	20
2.4-3	Trailing edge conditions	21
2.4-4	Digitizer timing	22
2.4-5	Sliding window detector	23
2.4-6	Mode S preamble detection timing	24
2.4-7	Mode S preamble detector	26
2.4-8	Pseudo and extra leading edge generators	27
2.4-9	Pseudo and extra leading edge generation	28
2.4-10	Pseudo leading edge generation, PW = 6	29
2.4-11	Pseudo leading edge generation, PW = 10	29
2.4-12	Extra leading edge timing	30
2.5-1	eply and interrogation controller (RIC) block diagram	32
2.5-2	Block Diagram of Am2910	33
2.5-3	Microprogram field definitions of the signal processor	36
2.5-4	RIC control register field description	37
2.5-5	RIC task control field assignments	37
2.5-6	Status bit definitions	39
2.5-7	Special I/O address assignments	40
2.6-1	Mode S interrogation generator feed-out encoder	42
2.7-1	Mode S reply processor feed-out decoder	44
2.8-1	Mode C reply accumulator (CRA)	45
2.9-1	Computer subsysystem	47
2.9-2	Z8002 program memory address assignment	49
2.9-3	I/O address space assignment	50

ILLUSTRATIONS (CONT'd)

Fig.		Page
3.2-1	Helical representation of ATCRBS leading edge (LE) pulse stream	59
3.2-2	Byte definitions for request formats	62
3.2-3	Byte definitions for reply formats	63
3.3-1	Program flow of the Am2910 microprogram software	65
3.3-2	Mode C processing routine	66
3.3-3a	Mode S processing routine	67
3.3-3b	Mode S processing routine	68
3.3-4	Mode S reply processing subroutine	69
3.3-5	Squitter mode processing routine	71
4.0-1	Laboratory test configuration for GATCAS unit	73
6.1-1	Reply data from Mode S head-on encounter	80
6.1-2	CAS tracks from Mode S head-on encounter	81
6.2.1	CAS tracks from Mode S head-on encounter	82
6.2-2	Histogram of the number of targets in track	88
6.2-3	Track probability vs interference density	91

TABLES

Table		Page
2.2-1	GATCAS RF link power budget	12
6-1	ATCRBS mode performance	84
6-2	Probability of report evalauted for aircraft of interest	86
6-3	Probability of track evaluated for aircraft of interest	87
6-4	Probability of track vs. target density evaluated for aircraft of interest	90
A-1	GATCAS cost summary	A-2

1.0 INTRODUCTION

Lincoln Laboratory has supported the FAA in the development of Traffic Alert and Collision Avoidance System (TCAS) surveillance techniques for air-carrier use since 1975. In anticipation of an eventual interest in TCAS by the general aviation community, Lincoln Laboratory was tasked in 1979, as a part of its FAA-sponsored TCAS development program, to also explore TCAS surveillance techniques suitable in performance and cost for general aviation use. This report documents the results of this task; it describes the GATCAS equipment that was designed, built, and tested for this program and it provides surveillance performance data from bench tests and flight tests of this equipment.

Flight testing of the GATCAS equipment took place during the winter of 1981/1982, and the work was concluded in the Spring of 1982. During the period in which the design specifications for this experimental GATCAS equipment were being developed, significant design concepts were still emerging, many of which were ultimately embodied in the National Standard for TCAS II. The GATCAS equipment that was flown included very few of these concepts and therefore must be considered strictly as an experimental design which was developed to better understand techniques for reducing the cost of critical surveillance elements associated with the TCAS transmitter and the reply processor. The equipment did not include ATCRBS or Mode S surveillance tracking algorithms. It included no interference limiting, bearing estimation, or CAS logic. It was flight tested by interfacing it with an existing air-carrier TCAS Experimental Unit (TEU) which did include ATCRBS and Mode S surveillance algorithms, CAS logic and recording capability, but no interference limiting, bearing estimation, or traffic advisory display logic.

Early paragraphs of Section 2. summarize the GATCAS surveillance design philosophy, list the performance requirements established, and describe the major architecture of the GATCAS avionics subsystems. Succeeding paragraphs in Section 2 describe the design of each of the subunits included in the GATCAS avionics. Section 3 describes the software developed to perform the real-time processing tasks. Diagnostic hardware and software are described in Sections 4 and 5 respectively. Flight test results obtained by interfacing to an existing TEU are presented in Section 6., and Appendix A is an analysis of the probable costs of GATCAS equipment based upon the design concepts employed in the experimental model.

2.0 GATCAS DESIGN

2.1 Overview

2.1.1 Design Philosophy

The goal of the GATCAS effort was to determine if the more benign air traffic environment in which most general aviation TCAS equipment would be expected to operate (relative to the higher-speed, higher-altitude air carrier TCAS environment), did in actuality permit the lower transmitter power levels, the reduced reply processing requirements, and the simplified design necessary to reduce the cost of GATCAS avionics when produced in quantity.

To determine the performance requirements for GATCAS equipment a maximum operating ceiling of approximately 10,000 feet was accepted as the target for the class of aircraft to be equipped. Range-speed statistics collected for aircraft below 10,000 ft. during extended flights in the Los Angeles Basin in 1979 (see 42QTL-BCAS-4-12, pages 25-28) showed that the measured closing rate of more than 99% of the aircraft encountered did not exceed 300 knots. A value for pilot warning time (Tau_R) of 25 seconds was adopted as appropriate for closing rates of less than 300 kt based upon previous CAS experience. Assuming a track acquisition time of 10 secs and a fixed warning range of 0.5 nmi, the required maximum surveillance range is thus

$$R_{max} = \frac{300 \text{ nmi/hr}}{3600} \times 35 \text{ sec} + 0.5 \text{ nmi} = 3.42 \text{ nmi}$$

The comparable range for an air-carrier TCAS capable of 25-sec warning of 1200-kt closing rates with a 1-nmi fixed warning range is 12.67 nmi. The shorter range requirement for GATCAS allows one to reduce the transmitted power requirement from 500 watts to approximately 50 watts. This, together with a related reduction in power supply capacity, would be the basis for a significant reduction in cost.

The reduced range requirement also reduces the average number of synchronously garbling replies expected in response to each ATCRBS interrogation. If the GATCAS could be made to successfully degarble up to 3 overlapping replies, it could operate with track densities of 3 aircraft within in 3.42 nmi (or approximately 0.08 aircraft per nmi^2) without the need for whisper-shout and its costly digital microwave attenuator.

Parallel developments had made it apparent that the ATCRBS reply processing and degarbling circuitry, implemented in hardware up to that time, could be implemented in software much more economically. The reduced target counts in GATCAS reinforced the decision to develop this new reply processing approach. This aspect of the GTCAS effort would also be applicable to cost reduction in air-carrier TCAS equipment. For this reason, the development of a software reply processor was considered a major goal of the program.

2.1.2 System Operating Characteristics

As noted above, the GATCAS avionics was intended to provide air-to-air surveillance for general aviation aircraft normally operating below 10,000 feet. The operating characteristics selected for the GATCAS experimental unit are summarized as follows:

Peak Transmit Power (at RF port):	100 watts
Receiver Sensitivity (MTL):	-72 dBm
Maximum processed	8.7 nmi
ATCRBS Mode Track Capacity (aircraft)	
Processed and Degarbled:	5
Mode S Mode Track Capacity (aircraft)	
Active:	5
Dormant:	12
Squitter:	25

These characteristics provided somewhat more capability than is strictly required for the GATCAS function as summarized in para. 2.1.1. However excess capability was included in this experimental test bed to provide additional experimental flexibility. It was possible to degrade the performance to match the ultimate design goals intended for this class of equipment. The analysis of the surveillance flight test data obtained with this equipment (see Section 6) focused on the performance of the equipment at reduced target ranges.

2.1.3 Major System Elements

The GATCAS is a self-contained experimental avionics unit which includes a transmitter-receiver, a digital processor, and a 16-bit computer (Fig. 2.1-1). The computer subsystem is adequate to support surveillance and collision avoidance tasks, and to generate output for display to the pilot. However, to reduce the programming required to make the GATCAS unit functional, an existing air carrier TCAS experimental unit (TEU) was configured to accept reply data from a GATCAS unit serial I/O port so it could perform the real-time processing and recording using existing software. The GATCAS unit thus performed only interrogation control and reply processing in its functional tests. The operational system configuration is shown in Fig. 2.1-2.

The internal architecture of the GATCAS is shown in Fig. 2.1-3. The system is divided into two major components, the computer subsystem and the signal processor. The computer subsystem uses a Z8002 microprocessor and

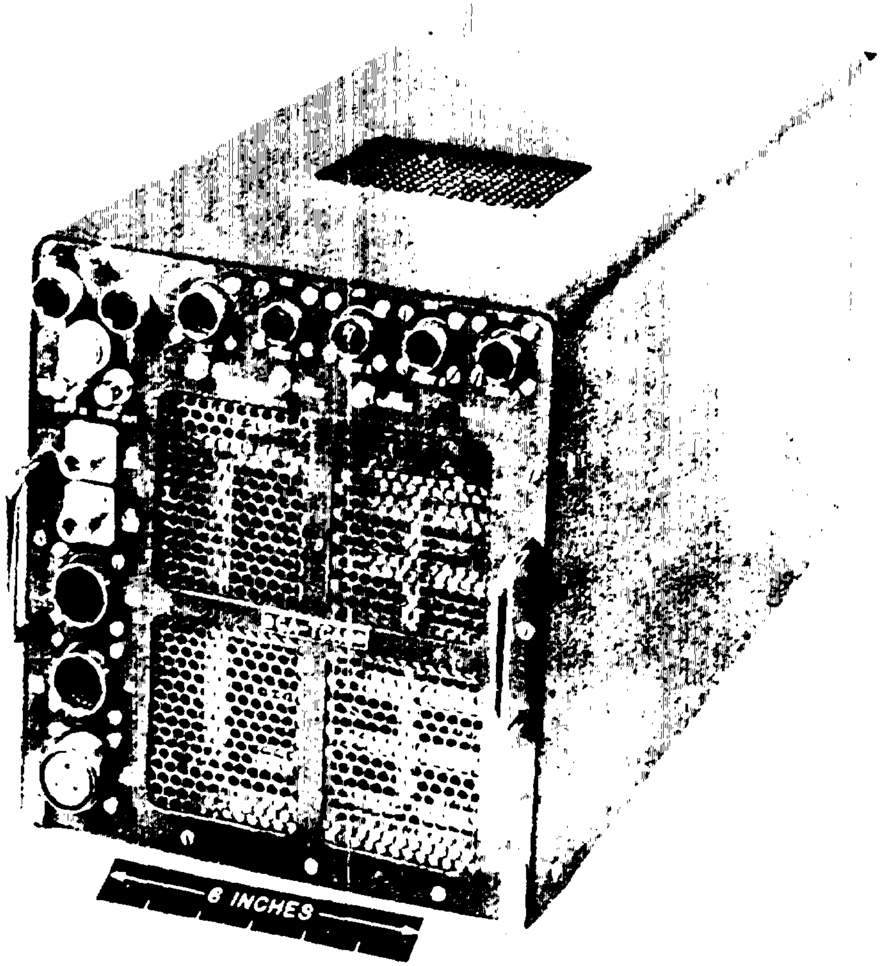


Fig. 2.1-1. GATCAS collision avoidance unit.

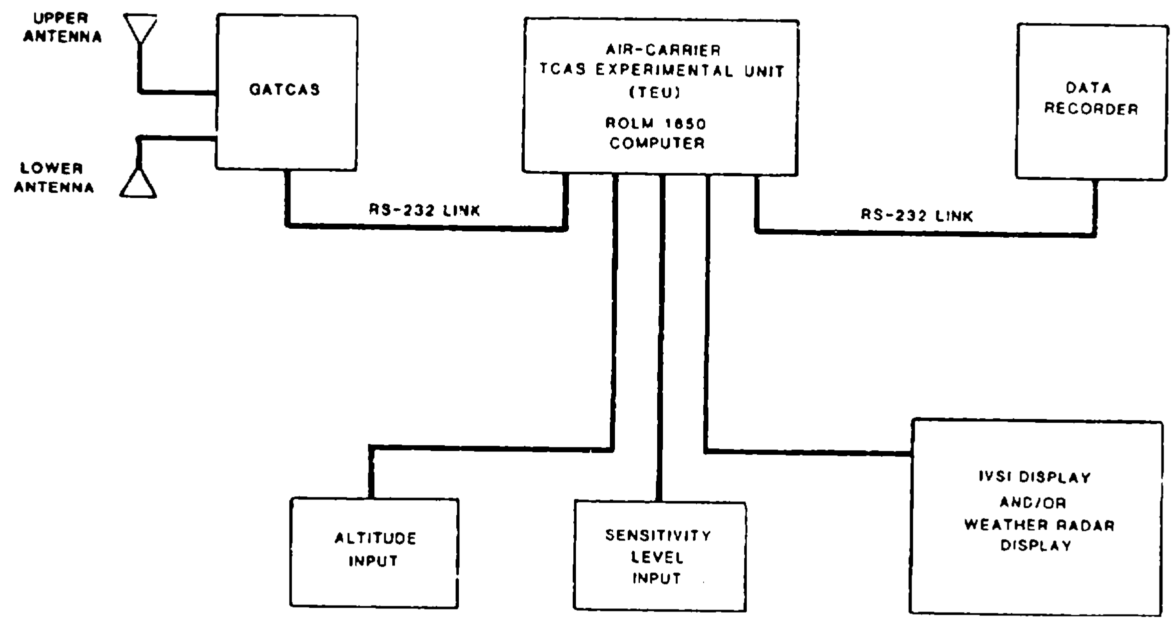


Fig. 2.1-2. GATCAS operational configuration.

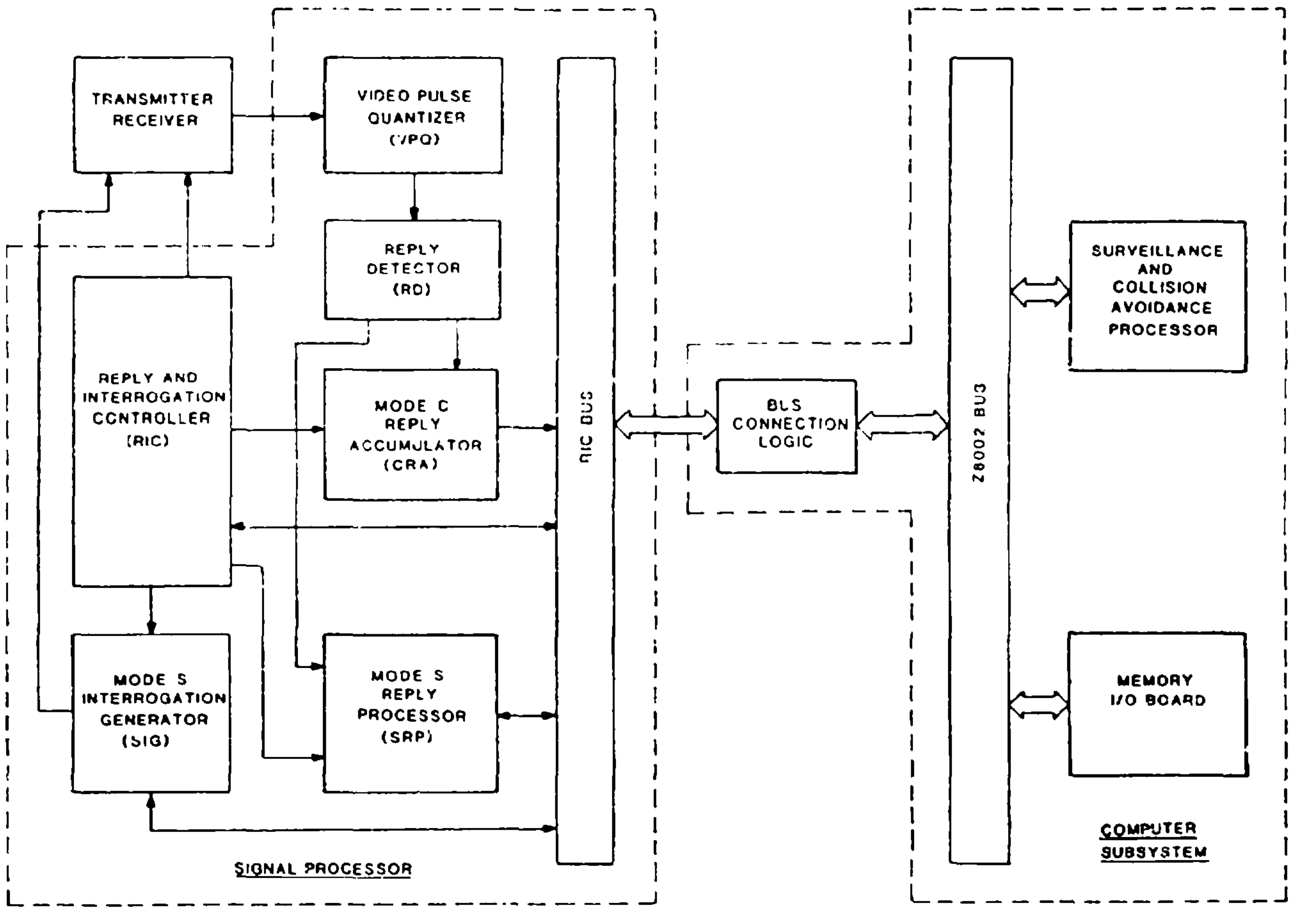


Fig. 2.1-3. GATCAS architecture.

ATC-115

provides for two parallel I/O ports, four serial I/O ports, a system timing controller, an interrupt controller and 191 Kbytes of memory. All parts of the computer subsystem are connected using one bus (Z8002 Bus).

The signal processor uses an Am2910 microprogrammed sequencer to control interrogations and replies in both the Mode S and Mode C modes. The signal processor has an internal bus (RIC bus) which connects the Mode S and Mode C memories, range counter, control register, and status register. The Z8002 can read and write to the appropriate registers of the signal processor through the use of bus connection logic which connects the RIC bus to the Z8002 bus.

2.2 RF Unit

The GATCAS RF Unit, Fig. 2.2-1, consists of the following functional subunits:

a) Microwave signal source

b) PAM/DPSK modulators

c) Microwave power module

d) RF components

e) Receiver

2.2.1 Microwave Signal Source (1030 MHZ)

The microwave signal source consists of a phase-locked, temperature-controlled crystal oscillator, stablized to within ± 0.0005% of the nominal frequency, and a solid state multiplier chain which generates the 1030 MHz transmitter frequency. The output power of the oscillator-multiplier chain is sufficient to permit the insertion of a circulator and a 10-dB pad between the source and the phase-shift modulator (double-balanced mixer). The isolation and the use of an absorptive (rather than reflective) pulse amplitude modulation (PAM) switch are provided to maintain the carrier frequency stability required by the TCAS National Standard.

The microwave source is located in a shielded enclosure to minimize 1030 MHz leakage to the co-located airborne transponder.

The same 1030 MHz frequency source is also used as the receiver local oscillator to convert the 1090 MHz replies to a 60 MHz intermediate frequency (IF).

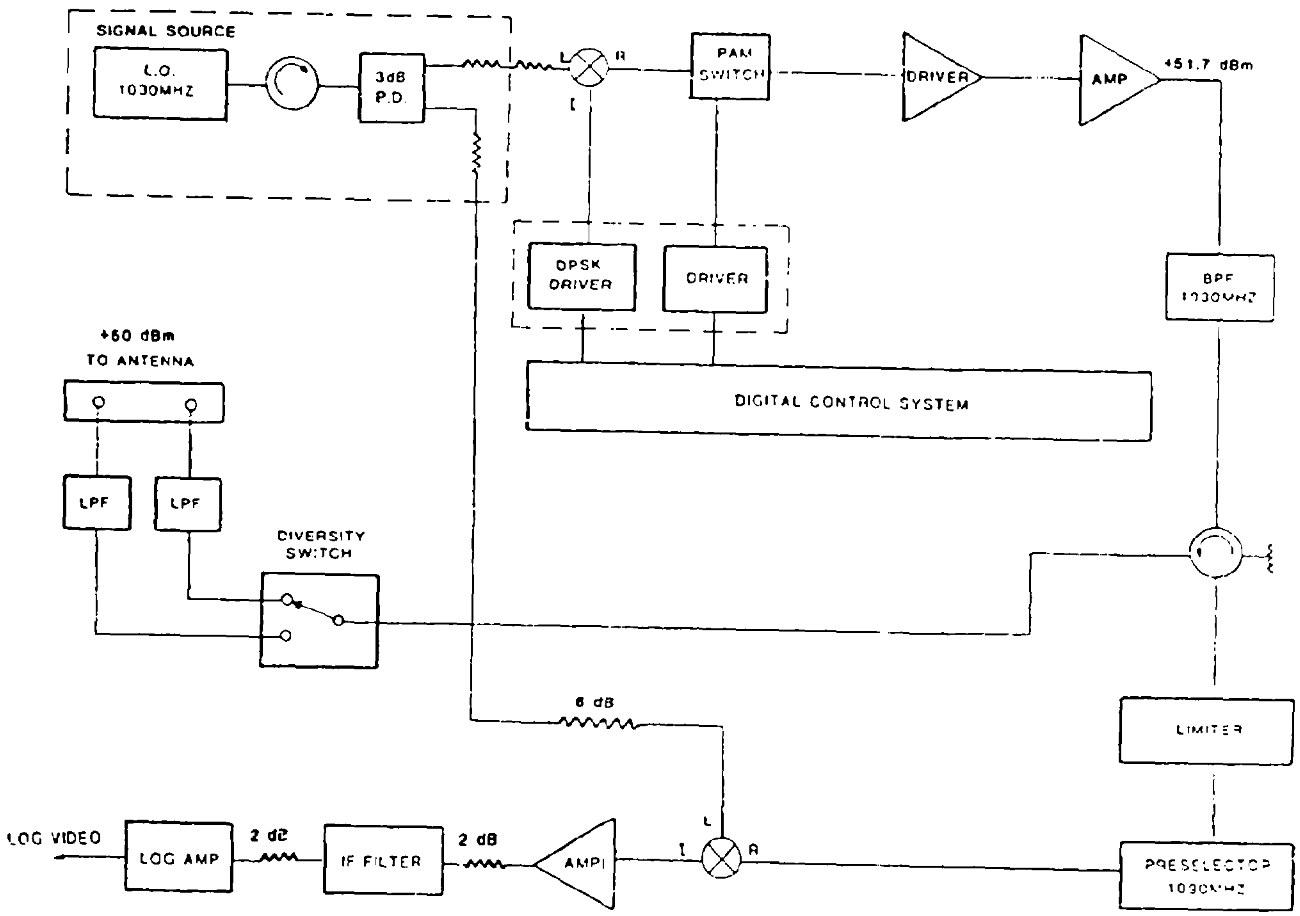


Fig. 2.2-1. General aviation TCAS microwave block diagram.

2.2.2 PAM/DPSK Modulators

Two types of modulators are required for the RF Unit: a pulse amplitude modulator (PAM), and a differential phase-shift keyed (DPSK) modulator. The PAM modulator consists of an absorptive type switch with an off/on ratio of 110 dB.

The DPSK modulator consists of a double balanced mixer, and a bipolar video driver. Since the bandwidth of the double-balanced mixer is large relative to the transmit signal bandwidth, the DPSK transition is determined by the rise-time of the driver's video pulse and by the transmitter bandpass characteristics. The Mode S National Standard requires that these transitions be completed within 80 nanoseconds.

2.2.3 Microwave Power Module

The microwave power modules consist of a microwave solid state driver that provides 30 dB of gain at 1030 MHz, and a solid state power amplifier that provides 22 dB of gain. The operating voltage required for the driver is +20 VDC; the operating voltages for the power amplifier are +33 VDC and +50 VDC. These voltages are derived from a common power supply which is energized by the aircraft 28 volt source.

Typical performance of the transmitter is shown in Fig. 2.2-2 which illustrates a transition of the DPSK modulation. It can be seen that the bandwidth of the transmitter is adequate to achieve the required 80 nsec transition time.

2.2.4 Other Transmitter RF Components

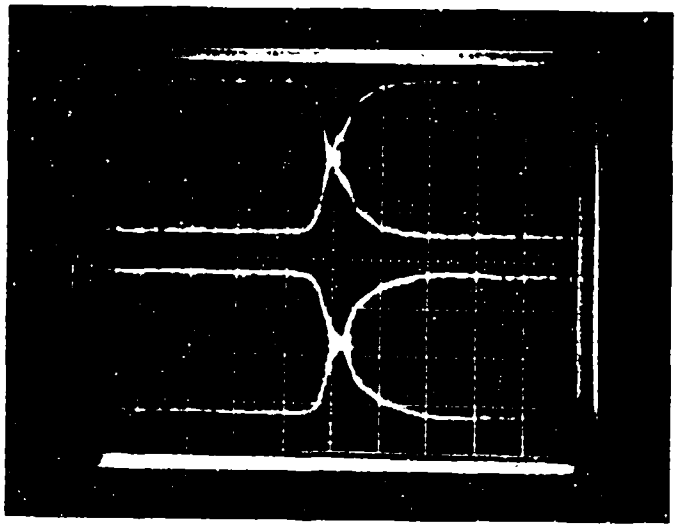
The transmitter also includes the following RF transmission and switching devices:

a) A transmitter output bandpass filter.- This filter, centered at 1030 MHz, has a bandwidth of 20 MHz, an insertion loss of 0.5 dB, provides 60 dB attenuation (with respect to band-center) at the receive frequency (1090 MHz), and limits the transmit spectrum in accordance with the Mode S National Standard.

b) A circulator.- This device acts as diplexer, connecting transmitter, receiver and antennas. It provides at least 25 dB isolation between transmitter and receiver.

c) A receiver input limiter.- This is used for receiver input protection. It can protect against 300 watt peak pulses.

d) A receiver input bandpass filter (preselector).- This filter is centered at 1090 MHz, has a bandwidth of 20 MHz, and provides 40 dB rejection to unwanted signals at the transmitter frequency (1030 MHz).



A

B

Hor. 50 nanosec/cm.

Fig. 2.2-2. Typical DPSK phase transition (180° phase reversal). (a) input to 150W amplifier, (b) output of 150 watt amplifier.

e) A diversity switch.- This solid-state SPDT switch connects either the top or bottom mounted antenna, as commanded by surveillance algorithms. It provides 25 dB of inter-channel isolation, and switches in 10 μsec.

f) Two antenna low pass filters.- These low pass filters prevent the radiation of harmonics generated by the diversity switch.

2.2.5 Receiver

The receiver consists of an image filter, a down-converter, a 60 MHz IF filter and a log amplifier.

a) Down-converter.- The insertion loss of the down-converter is 6 dB when +7 dBm LO power is injected to the L-port. The unit can withstand up to +26 dBm at room temperature (+17 dBm at 100°C). The 1 dB compression point occurs when the signal power level at the L-port is 1 mw.

b) IF amplifier.- This is a solid state, modular, wide-band amplifier.

c) 60 MHz IF filter.- This filter has a bandwidth of 10 MHz and establishes signal and noise bandwidth. It is a 6-pole Bessel filter designed to maintain phase linearity within its 3 dB bandwidth. Its insertion loss is 2 dB.

d) Log amplifier.- The 60 MHz signal is further amplified and video-detected using a miniature log amplifier. Its center frequency is 60 MHz and bandwidth is 20 MHz. Over an input range of -70 to 0 dBm its output rises from 0.2 to 2 volts into 93 ohms, a transfer characteristic of 25.7 mv/dB. The detected output video is fed to the receiver video monitor and the video pulse quantizer (VPQ).

Design of the receiver is based on the power budget and assumptions given in Table 2.2-1. Receiver performance is demonstrated in Fig. 2.2-3a,b,c which indicate a typical receiver log video response for the first two pulses of a Mode S reply preamble at -46 dBm.

2.3 Video-Pulse Quantizer

The Video Pulse Quantizer (VPQ) processes receiver log video pulses to produce quantized slope and signal strength, and a Mode S chip amplitude comparison signal. The VPQ is designed to produce, in conjunction with the pulse digitizer described in Section 2.4.3, accurate time-of-arrival and pulsewidth estimates for all received pulses over the full dynamic range of the receiver where pulsewidth and time-of-arrival are defined at the -6 dB points.

TABLE 2.2-1

GATCAS RF LINK POWER BUDGET

Transmitter power at RF unit port or transponder RF port	+50 dBm
Antenna cable losses (both aircraft)	-6 dB
Free-space loss at 5 nmi	-112.5 dB
Receiver Signal Level at Transponder or RF unit RF Port	-68.5 dB
Thermal noise in 10 MHz bandwidth	-104 dBm
Receiver component losses	5.9 dB
Mixer, conversion loss	6.0 dB
IF amplifier noise figure	2.5 dB
Receiver noise power	-89.6 dBm
Signal to Noise Ratio at 5 nmi	21.1 dB
MTL (-72 dBm) to Noise Margin	17.6 dB

a)
1087 MHz

b)
1090 MHz

c)
1093 MHz

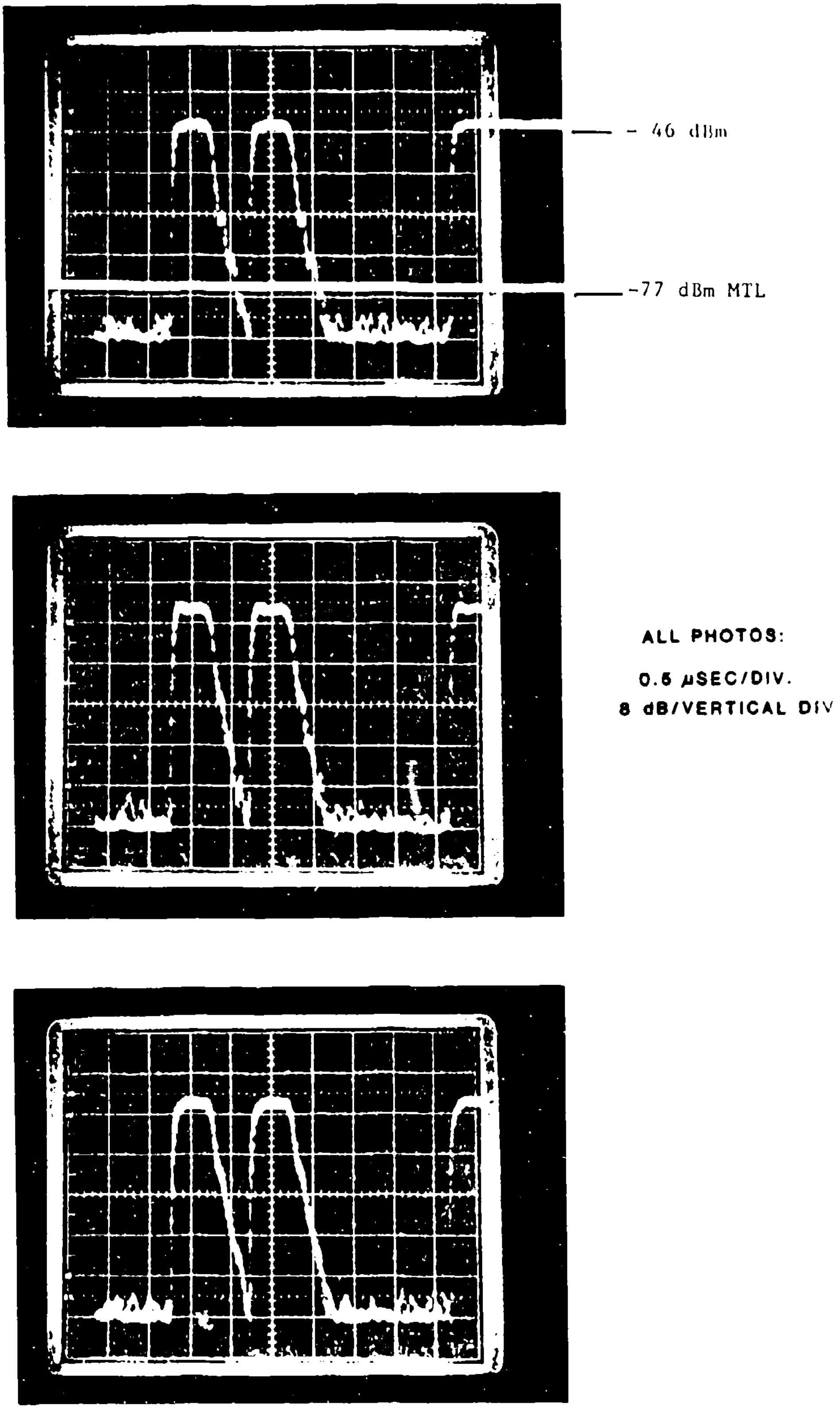


Fig. 2.2-3. GATCAS receiver log video response.

A simplified block diagram of the VPQ is shown in Fig. 2.3-1. Received log video pulses are amplified by A1 to produce a 40 mv/dB sensitivity at the A1 output. Slope detection is realized by subtracting a 125 ns delayed version of the reply pulse from an undelayed version. As shown in Fig. 2.3-2, slope thresholds may be defined (48 dB per sec) such that the outputs from comparators A6 and A7 contain edges which have fixed time relationships with the leading and trailing log video 6 dB points, independent of signal amplitude.

Log video signal strength is determined at A8 by comparing each pulse with a fixed mean threshold level (MTL) and, if selected, a dynamic mean threshold value (DMTL). The minimum fixed MTL is established at -72 dBm by a potentiometer at the input to A4. The threshold may however be increased (sensitivity decreased) by DAC-2 under software control.

Dynamic MTL is provided to reduce the effect of multipath and is used in two modes. When operating in the ATCRBS mode, DMTL is enabled continuosly. As shown in Fig. 2.3-3, any pulse that exceeds MTL by more than 12 dB (due to the hysteresis at the input to A9) will cause A9 to switch. The timing network will then direct the sample-and-hold module to sample the amplitude of that pulse and hold it for 22 microseconds. The effect of this is to raise the receiver threshold to a value 9 dB below the amplitude of the actuating pulse in the case of ATCRBS or 6 dB for Mode S (Fig. 2.3-4). If other pulses are received with amplitudes less than 12 dB above the new threshold, the DMTL will remain unchanged. The sample-and-hold output is offset by DAC-1 at A3. The offsets for Mode S and ATCRBS are independently switch selectable. The hysterisis is provided to insure that early low-level fruit does not capture the DMTL circuit.

When operating in the Mode S mode, DMTL is enabled only by a Mode S preamble detection. A1 functions as in the ATCRBS mode. The timing module however samples and holds the amplitude of all pulses detected by A1 for 700 nsec. When a preamble is detected, a separate timer continues the sample and hold (S/H) in the hold mode for 120 additional microseconds and enables the DMTL enable switch. Mode S DMTL is used to prevent erroneous confidence declarations due to multipath, but has no effect on the Mode S data detection.

Mode S PPM data are detected by an amplitude comparison of the first and second chip positions at A5.

2.4 Reply Detector

2.4.1 General Description

The reply detector digitizes the positive slope, negative slope, and quantized video signals from the Video Pulse Quantizer (VPQ) and produces digital data streams for both Mode S and ATCRBS message processing. However, digitizing both Mode S and ATCRBS data in the same hardware requires the ability to select the clock frequency to be either 8.0 MHz or 8.27 MHz. To

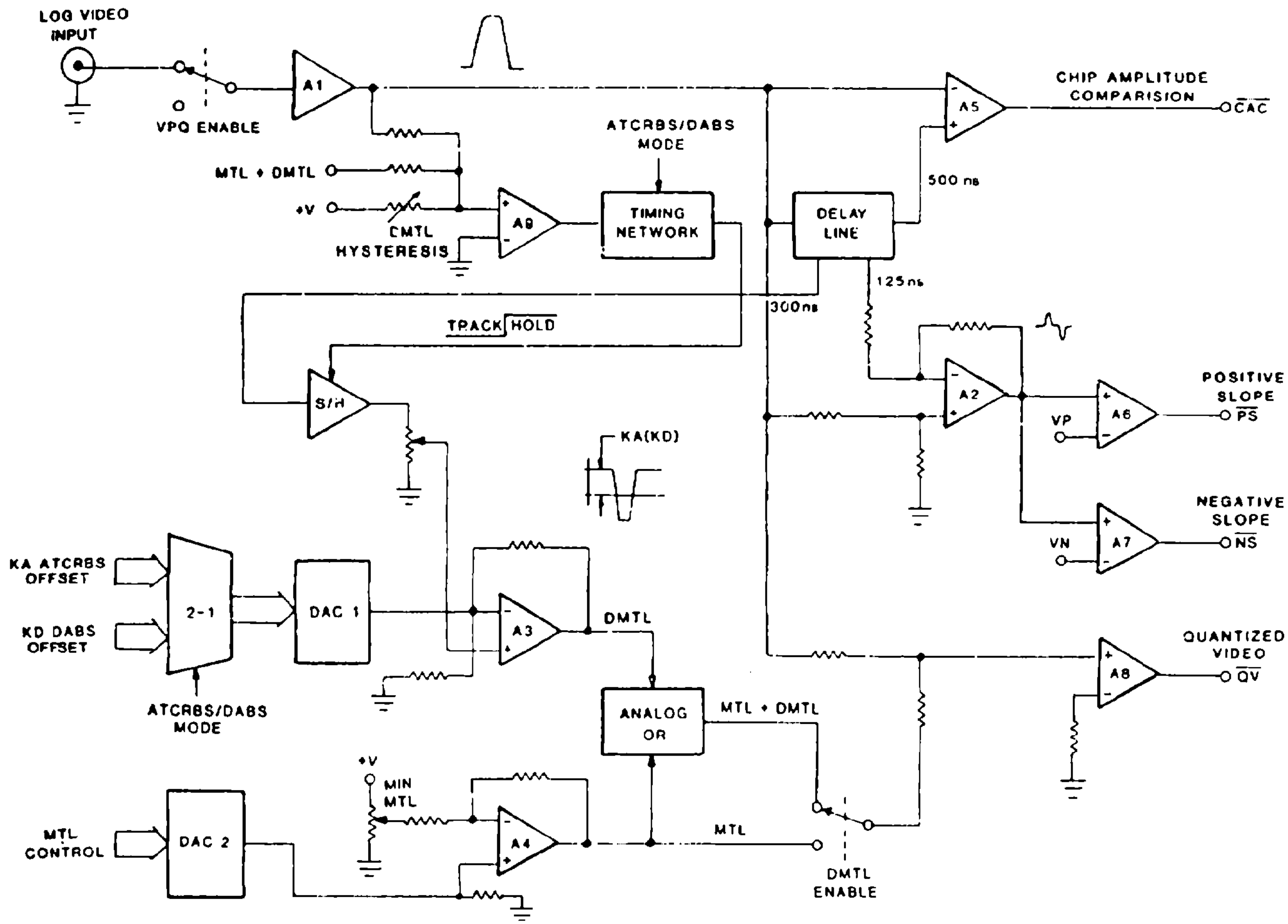


Fig. 2.3-1. VPQ block diagram.

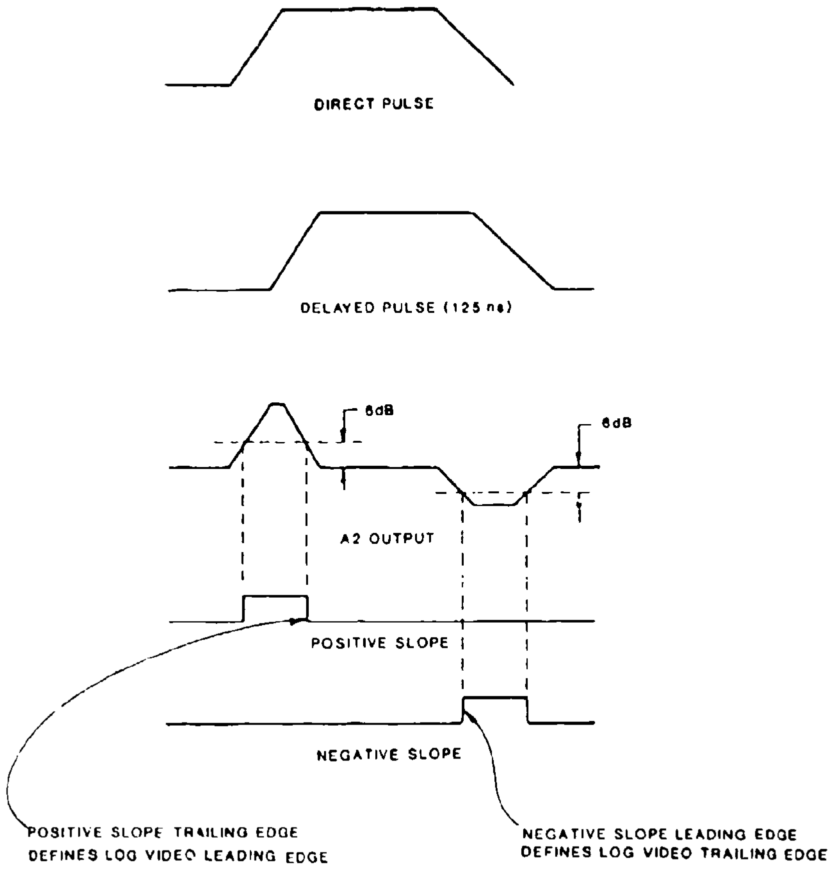


Fig. 2.3-2. VPQ waveforms.

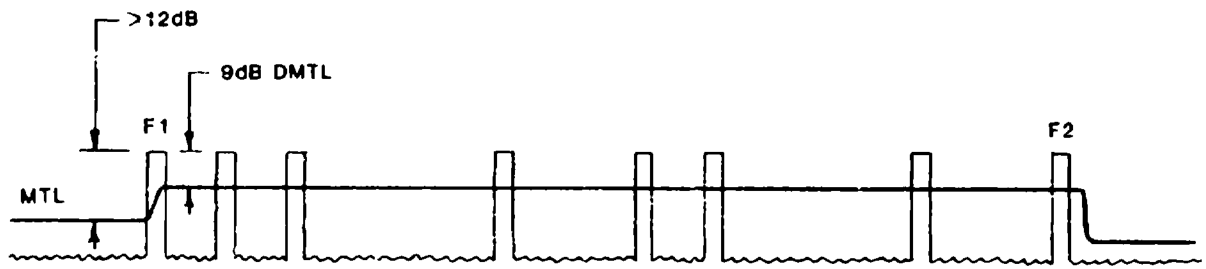


Fig. 2.3-3. ATCRBS DMTL.

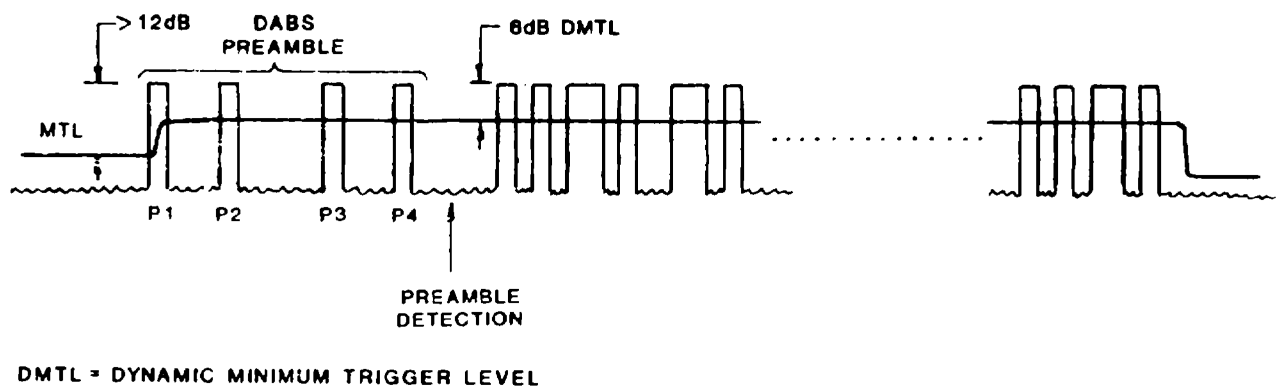


Fig. 2.3-4. Mode S DMTL.

accomplish this under RIC control, the system clocks (8.0 MHz or 8.27 MHz) drive a digitizer clock control circuit which selects the clock supplied to the digitizier. In addition to digitizing incoming signals, the reply detector also has circuitry to detect Mode S preambles and perform pseudo leading edge generation and short pulse rejection for ATCRBS replies. The 8.0 MHz, 8.27 MHz and 4.0 MHz clocks used throughout the system are derived in the reply detector.

2.4.2 Digitizer Clock Control

The digitizer operates using an 8.27 MHz clock frequency to process Mode S replies. The clock frequency is selected by bit 6 in the control register (section 2.5.5). When this bit is a 1, the 8.27 MHz clock is selected for ATCRBS mode operation and when the bit is 0, the 8.0 MHz clock is selected for Mode S mode operation. Figure 2.4-1 shows the clock controller and associated timing.

2.4.3 Video Digitizer

The video digitizer accepts slope and quantized video information from the VPQ and generates Actual Leading Edges, Actual Trailing Edges, and Sampled Quantized Video.

The digitizer logic circuit shown in simplified form in Fig. 2.4-2, establishes actual leading edges whenever PS makes a 1→0 transition and QV=1. This defines the leading edge of the log video pulse (~3 to 6 dB point depending on the specific video response which will vary slightly with reply frequency).

Trailing edges are defined in several ways in order to account for garble situations (see Fig. 2.4-3). When operating in the Mode S mode, an additional sixth class of trailing edge declaration, concerned with cusp phenomena is evoked (see Fig. 2.4-3e).

The generation of SQV for a single ungarbled pulse is shown in Fig. 2.4-4. The width of SQV matches the width of the log video pulse at the 6 dB width (variations will occur due to variations in the reply frequency which effects the pulse response of the receiver IF filter).

2.4.4 Sliding Window Detector

Mode S preamble detection requires that pulse energy be estimated. This is realized using a sliding window detector (SWD) which continuously examines SQV and notes when SQV=1 in at least 3 of the 4 samples in the window. The logic diagram and timing for P4 is shown in Fig. 2.4-5 and Fig. 2.4-6 respectively. Note that the preamble detector is enabled at the gate which generates the valid pulse signal (i.e., the input to the preamble detector). This is so that invalid data (multipath) does not enter the preamble detector following a message processing interval and synthesize a false preamble detection. Note also that the sliding window detector is enabled whenever the Mode S mode is selected in the control register.

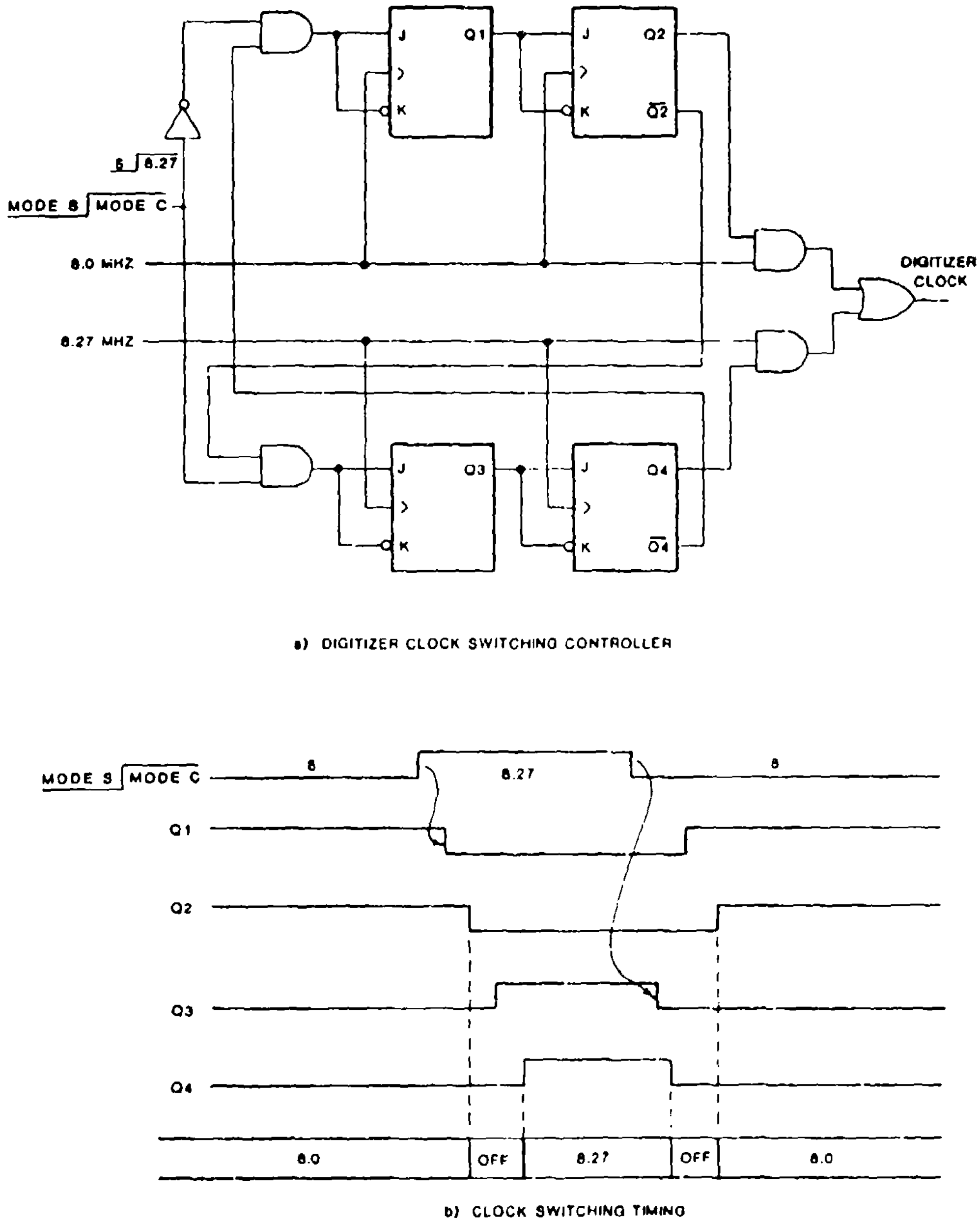


Fig. 2.4-1. Digitizer clock control.

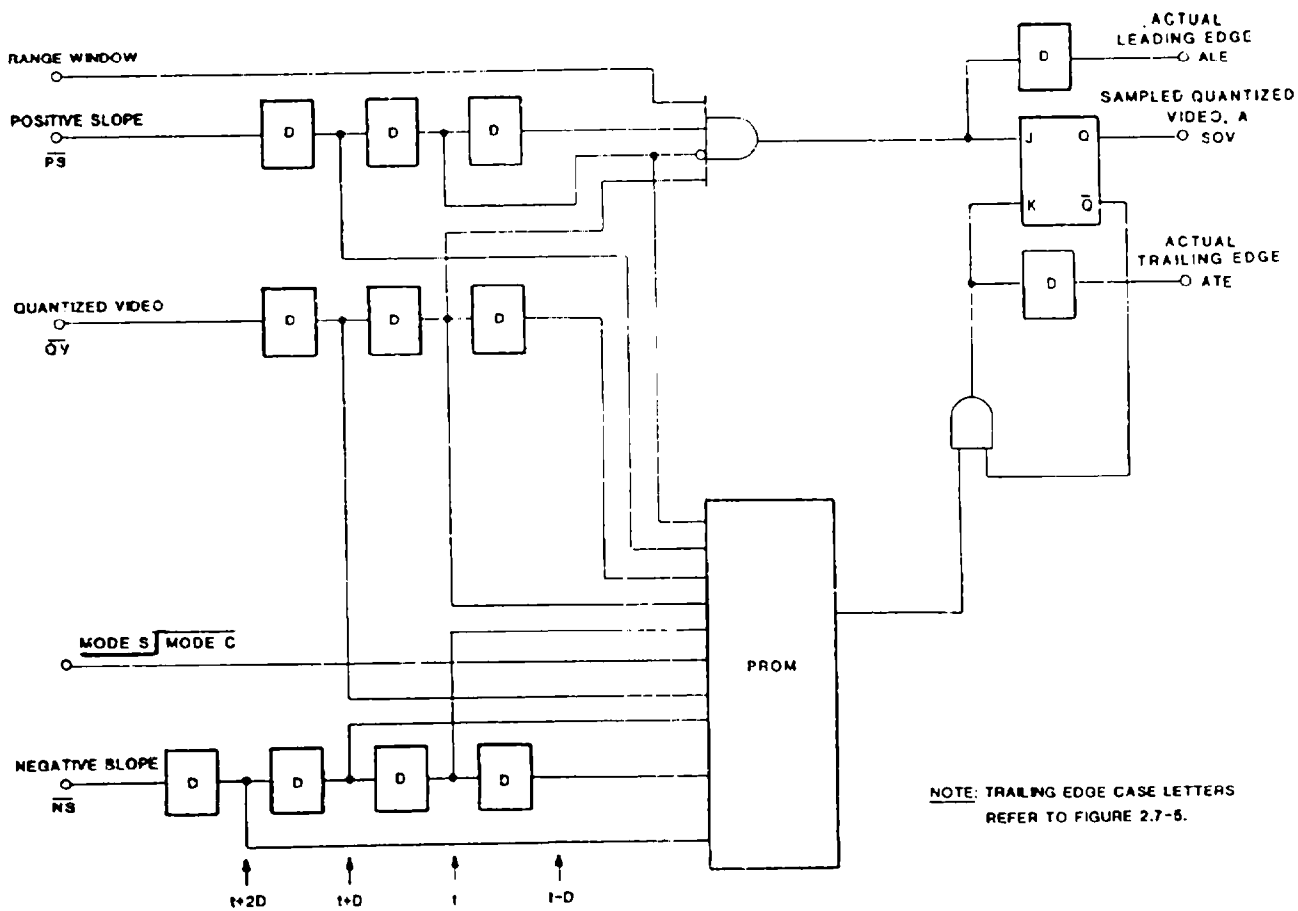


Fig. 2.4-2. Digitizer logic.

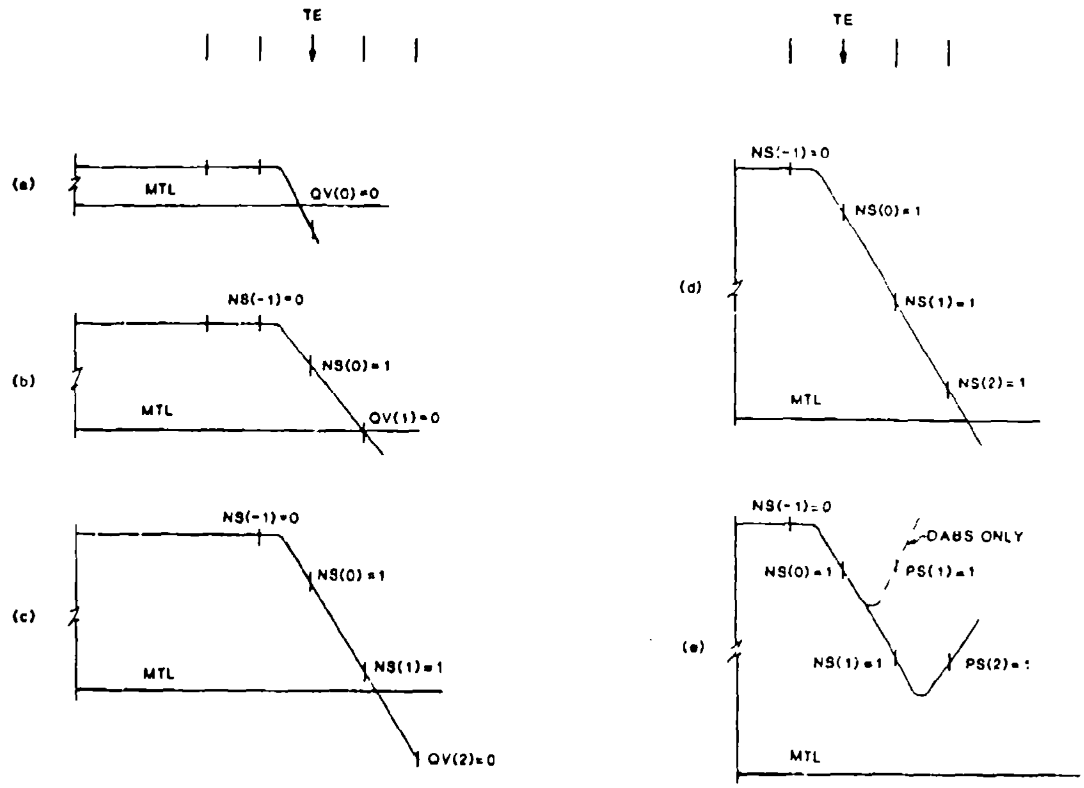


Fig. 2.4-3. Trailing edge conditions.

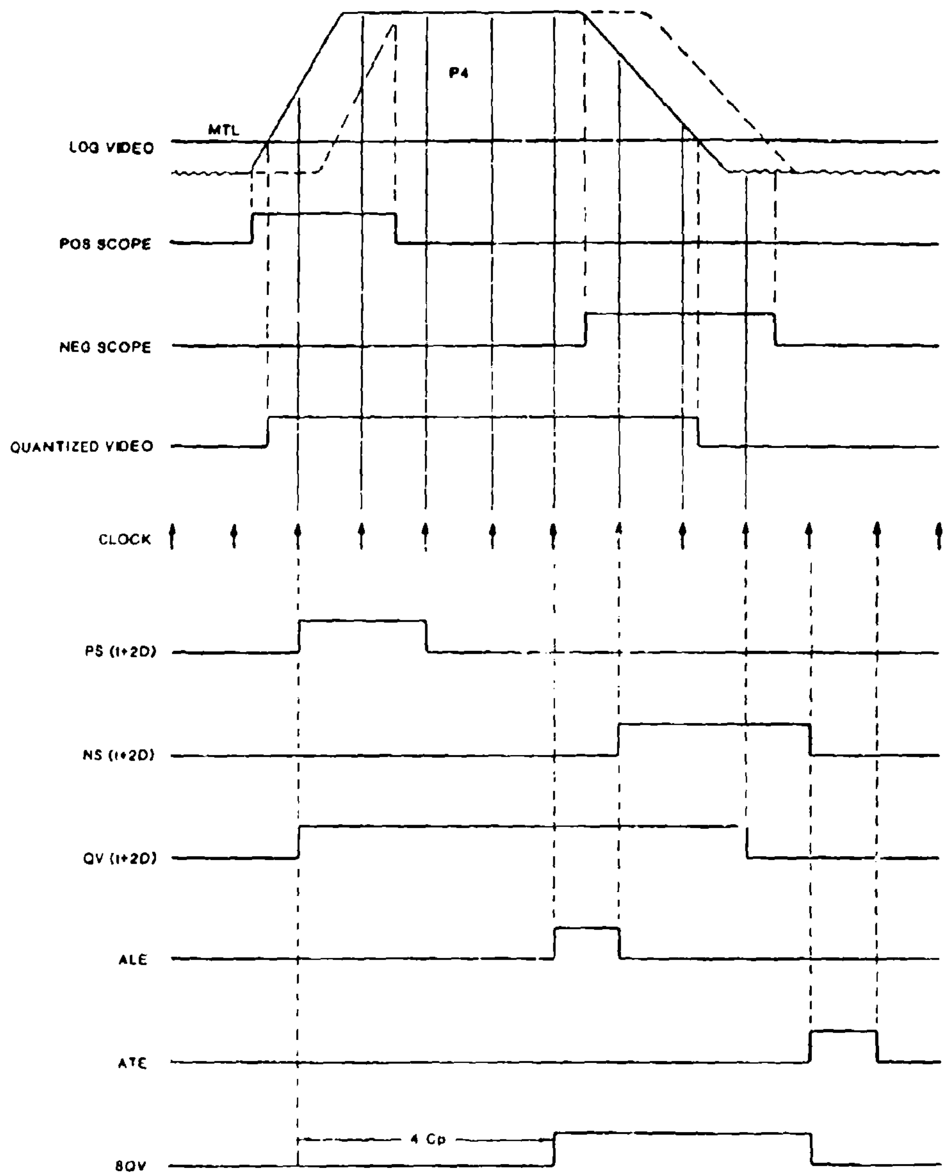


Fig. 2.4-4. Digitizer timing.

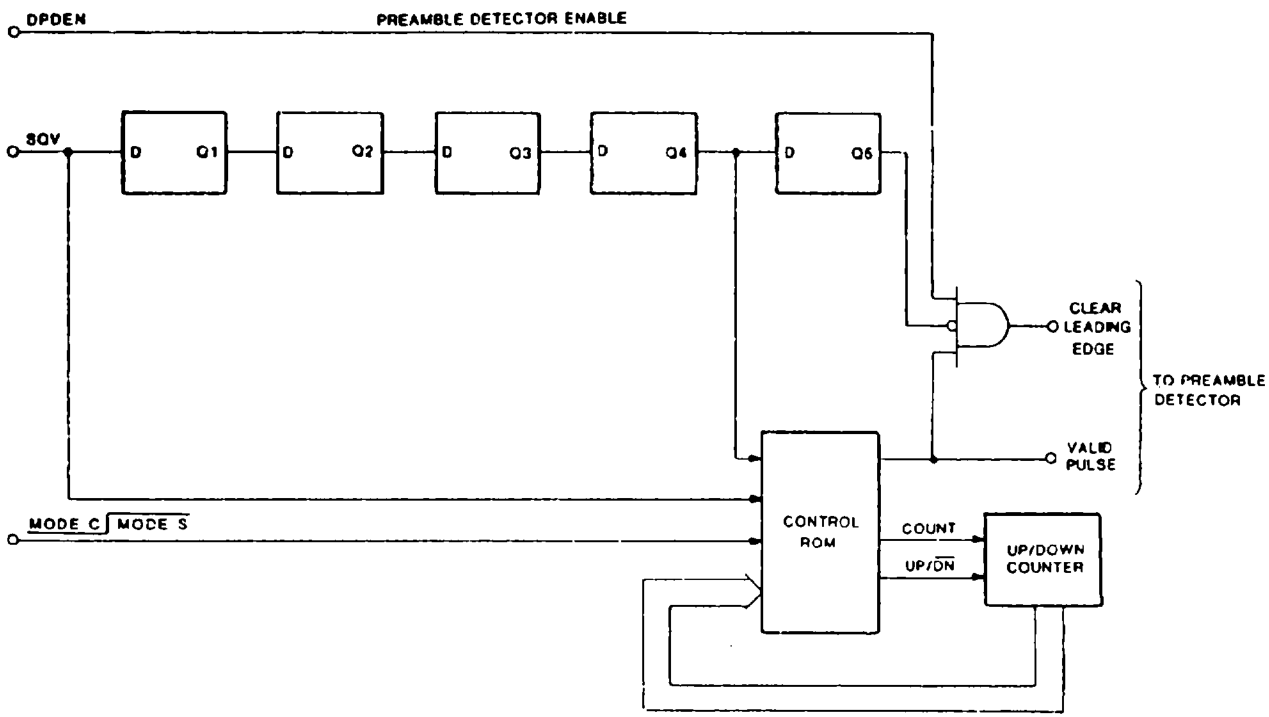


Fig. 2.4-5. Sliding window detector.

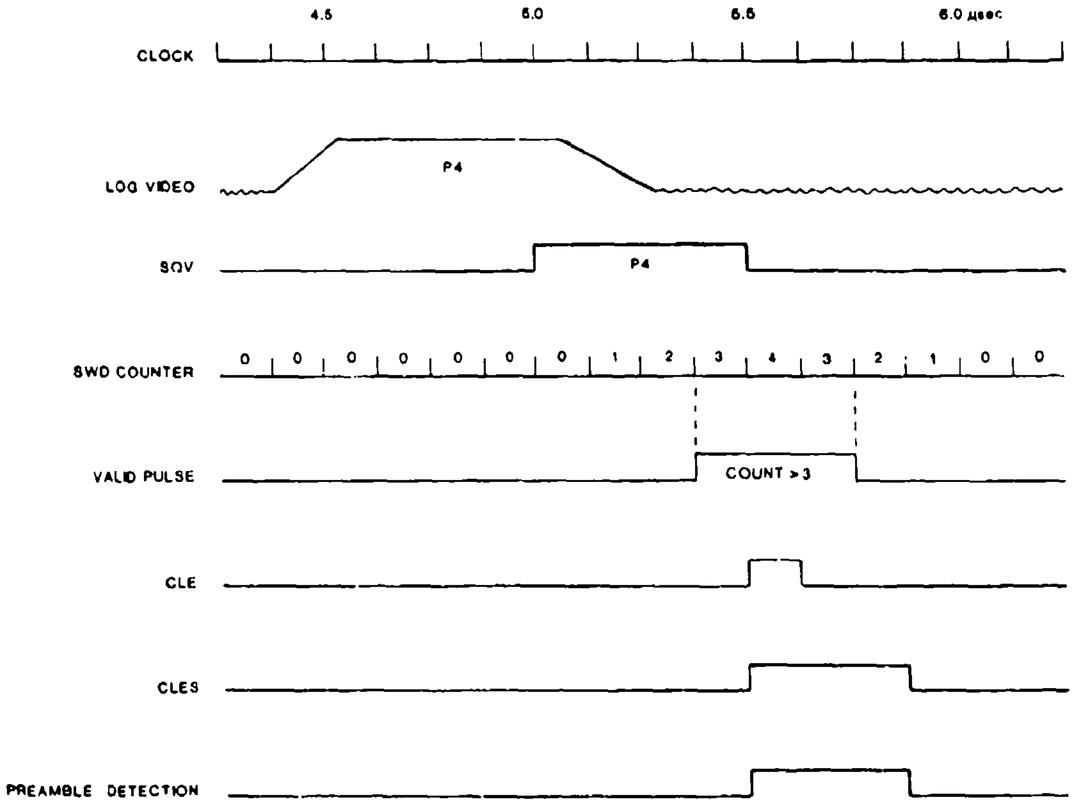


Fig. 2.4-6. Mode S preamble detection timing.

2.4.5 Mode S Preamble Detector

The preamble detector is shown in Fig. 2.4-7. It is enabled when valid pulses are enabled at the sliding window detector (see section 2.4.4). Two D flip-flops spread the CLE signal to account for timing tolerances. Figure 2.4-6 shows pertinent timing. Note that the detection of a preamble starts the Reply and Interrogation Controller (RIC) processing a Mode S reply and any further interrupts while the message is being processed are ignored.

2.4.6 Pseudo Leading Edge Generation and Short Pulse Rejection

Overlapping ATCRBS replies will generate pulses having widths different than the standard code pulse width (3-5 samples at 120.6 nanoseconds per sample). To reject narrow pulses and to estimate the leading edge positions of pulses that have been combined in the TEU receiver, the logic shown in Fig. 2.4-8 is used.

The first pulse processing step eliminates narrow pulses having widths less than three samples. Next, Counter 1 is used to artificially inject a pseudo leading edge four sample positions prior to the end of all pulses having widths greater than or equal to 6 (see figures 2.4-9, 2.4-10, and 2.4-11).

Finally, longer pulses are analyzed by the rest of the logic in Fig. 2.4-8 in order to insert extra leading edges every four samples starting at the leading edge of the pulse until the pseudo leading edge position is reached. The timing for a pulse having an 11 sample width is shown in Fig. 2.4-12.

A summary of the rules, adapted directly from the Mode S ground sensor design, for pseudo and extra leading edge generation is given below.

(a) If PW = 1 or 2, then the leading edge shall not be represented in the leading edge data stream, but all other directly declared leading edges shall be represented in the leading edge data stream.

(b) If PW ≥ 6, then a pseudo-leading edge shall be declared at the sample time four sample intervals prior to the declared trailing edge.

(c) If PW > 10, then additional pseudo-leading edges (called extra leading edges) shall be declared every fourth sample interval following the leading edge but prior to the pseudo-leading edge inserted in (b).

(d) If LW > 5, then additional pseudo-leading edges shall be declared every fourth sample interval following the first leading edge but prior to the second leading edge.

Note that when a leading edge declaration is followed by a trailing edge declaration, PW is used to denote this pulse width in terms of sample intervals. LW is used to denote the spacing between two successively declared leading edges in terms of sample intervals.

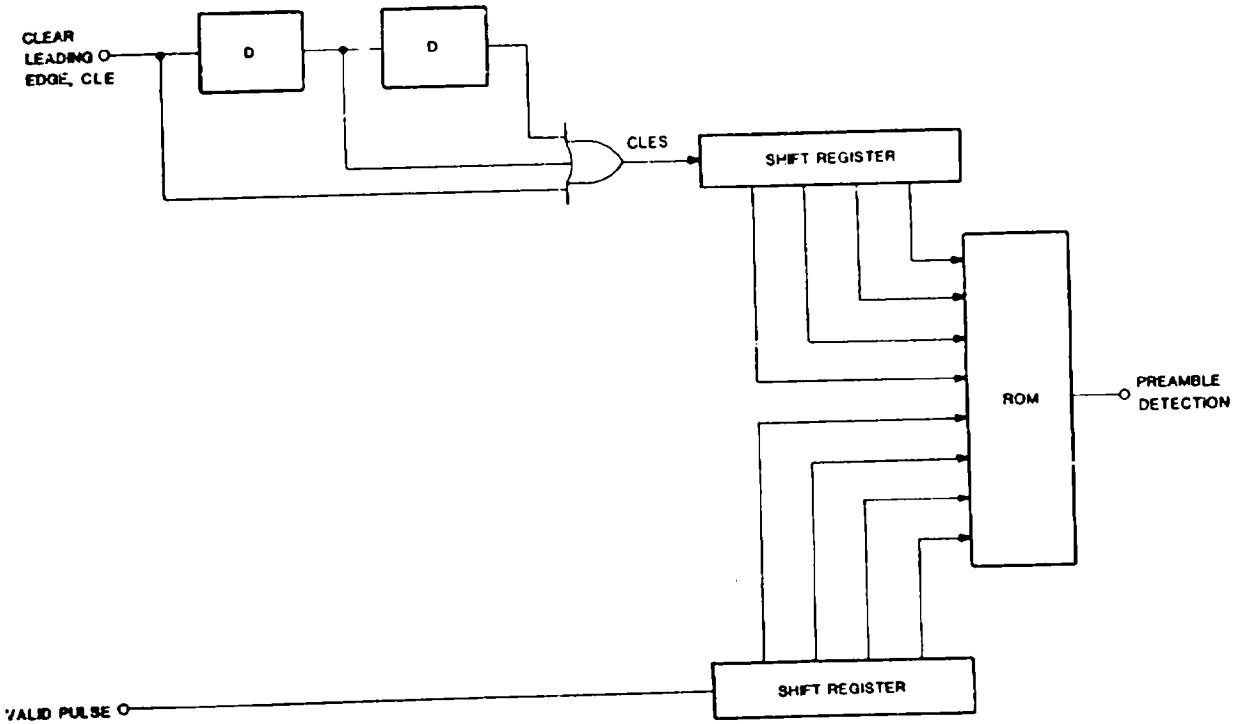


Fig. 2.4-7. Mode S preamble detector.

ATC-115

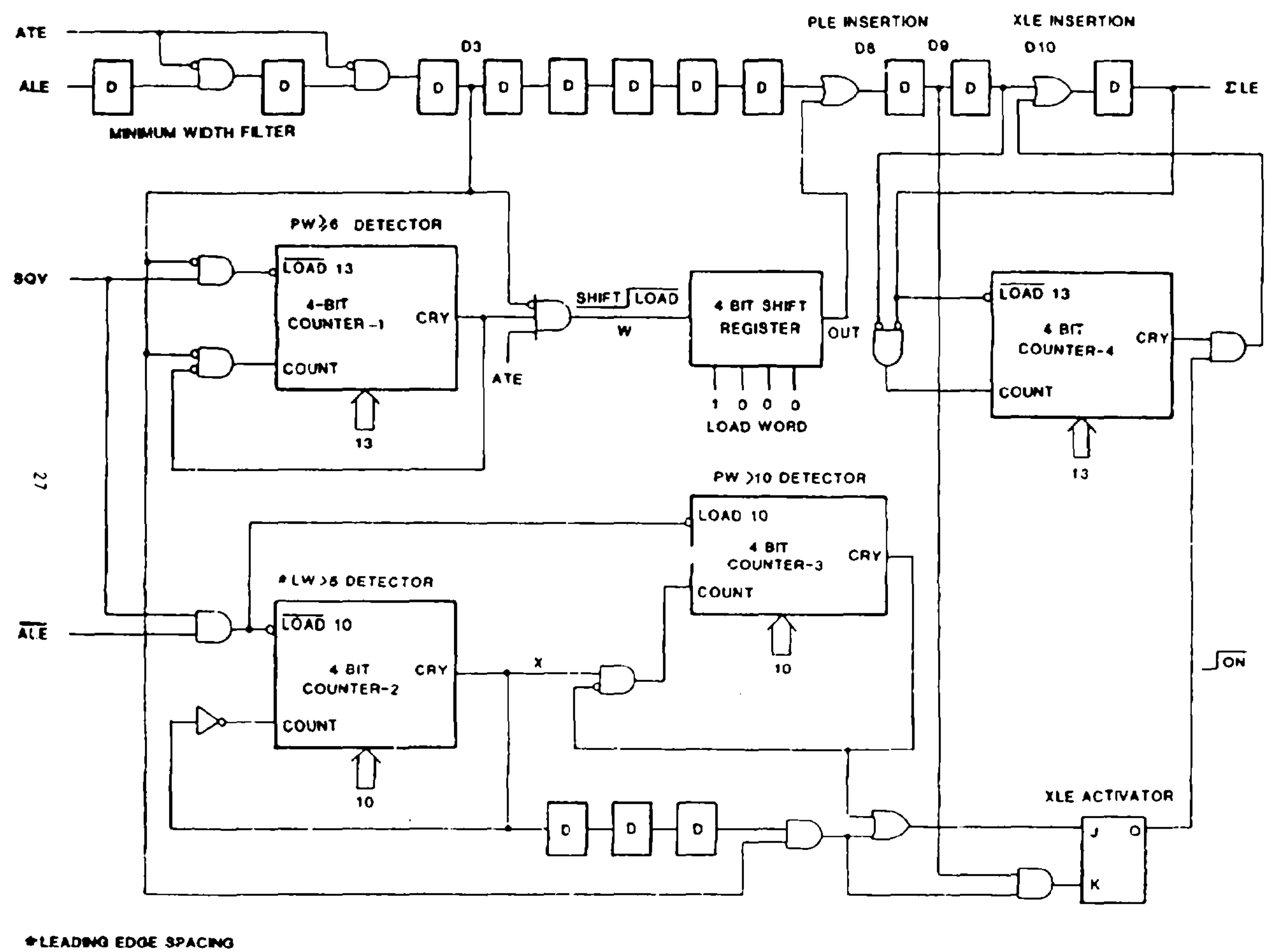


Fig. 2.4-8. Pseudo and extra leading edge generators.

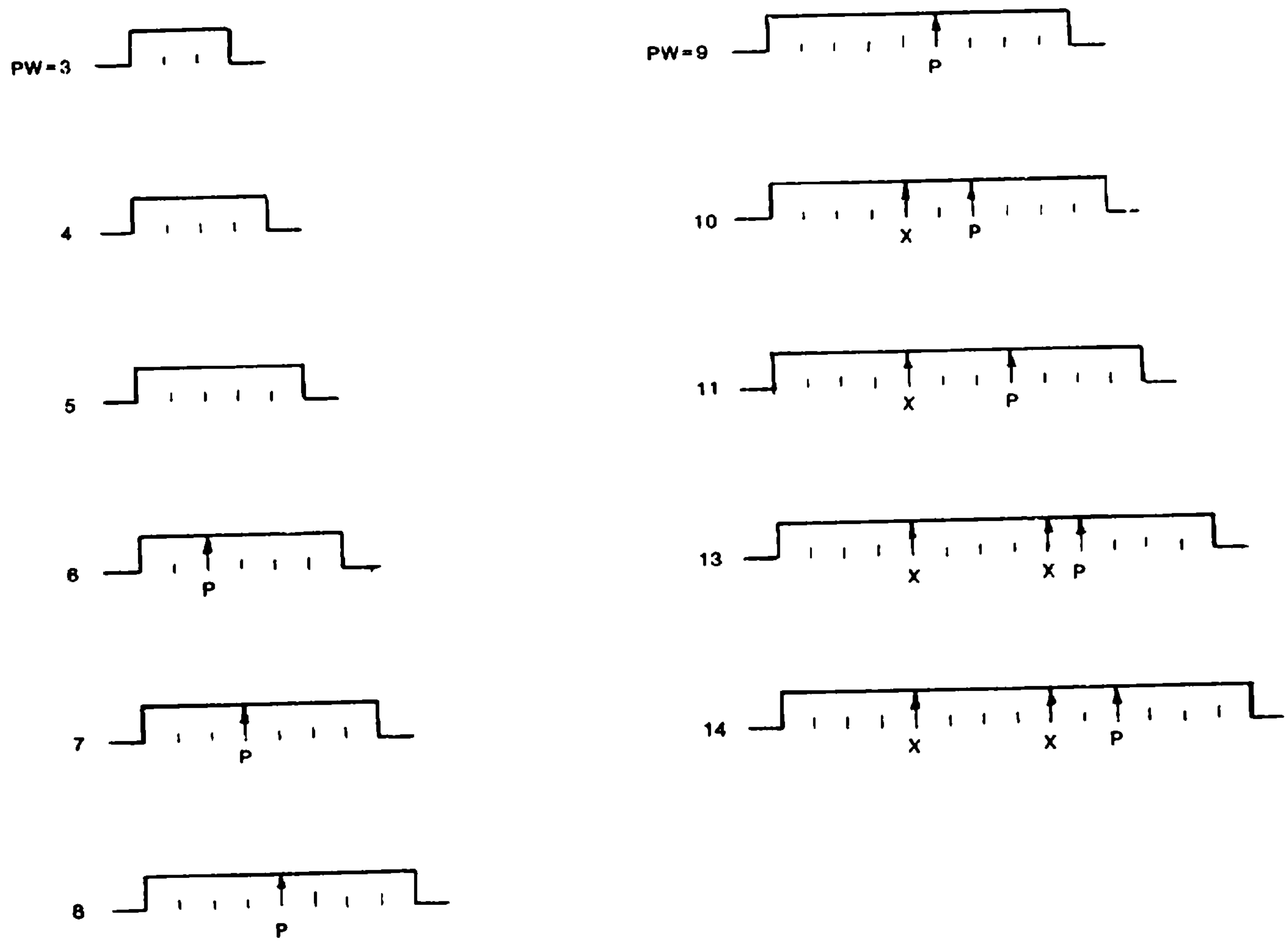


Fig. 2.4-9. Pseudo and extra leading edge pulse generation.

ATC-115

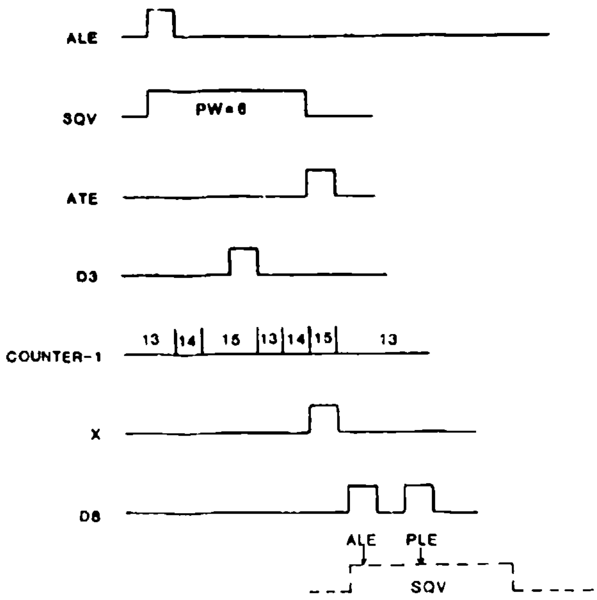


Fig. 2.4-10. Pseudo leading edge generation, PW = 6.

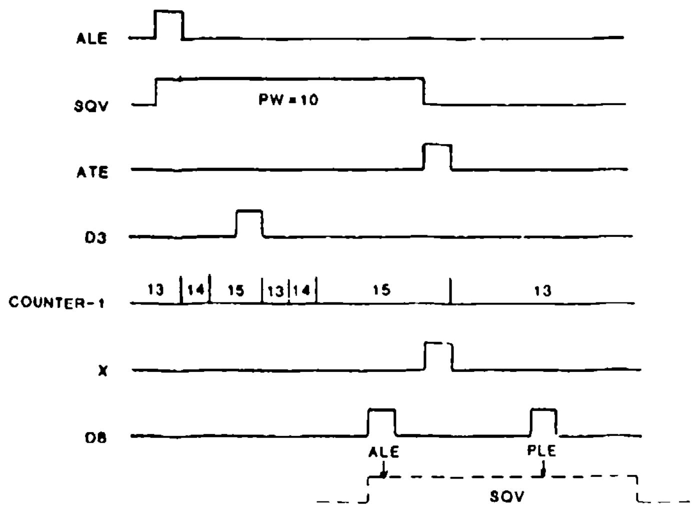


Fig. 2.4-11. Pseudo leading edge generation, PW = 10.

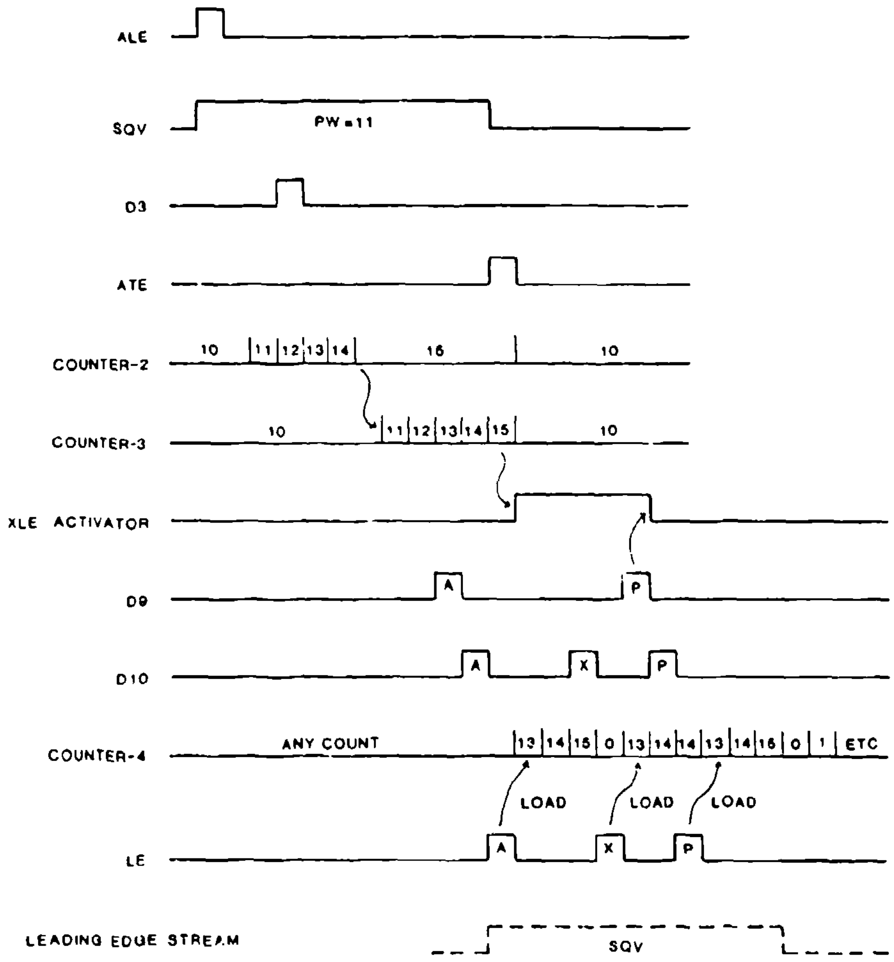


Fig. 2.4-12. Extra leading edge timing.

2.5 Reply and Interrogation Controller

2.5.1 General Description

The Reply and Interrogation Controller (RIC) controls the processing of Mode S and ATCRBS interrogations and replies. Upon command from the Z8002 computer, the RIC initiates and coordinates the execution of one of nine predefined tasks. Upon completion of one of these tasks, the RIC interrupts the computer so the results can be retrieved and processed.

The RIC consists of an Am2910 microprogram sequencer, Am2914 interrupt controller, microprogram memory, control register, status register, range counter, and Mode S data memory. A block diagram of the RIC is shown in Fig. 2.5-1.

2.5.2 Am 2910 Microprogram Sequencer

The Am2910 is a bipolar microprogram controller intended for use in high-speed microprocessor applications [1]. Its internal architecture has a fixed-width, 12-bit, data path allowing for an address space of up to 4K words of microprogram. Figure 2.5-2 shows a block diagram of the internal architecture. The controller contains a four-input multiplexer that is used to select either the register/counter, microprogram counter, direct input, or stack, as the source of the next micro-instruction address.

The register/counter consists of 12 D-type, edge-triggered flip-flops, with a common clock enable. The data bus furnishes data for loading the register/counter.

The microprogram counter consists of a 12-bit incrementer followed by a 12-bit register. An external control to the incrementer allows the microprogram address to be incremented or to remain unmodified. If the microprogram address is not incremented, the same micro-instruction is executed any number of times. This feature is critical in properly synchronizing the signal processor to decode a Mode S reply.

The third source for the multiplexer is the direct (D) input. This is used when a branch is required to alter program flow.

The fourth input to the multiplexer is a 5-word by 12-bit stack. The stack is used to provide return address linkage when executing micro-subroutines or loops. The stack contains a built-in stack pointer (SP) which always points to the last file word written. The stack pointer operates as an up/down counter.

The internal parts of the Am2910 are controlled by the instruction programmable logic array (PLA) which is driven by control signals and instruction inputs from the microprogram memory and an external conditional input. The instruction executing the external conditional input to the Am2910

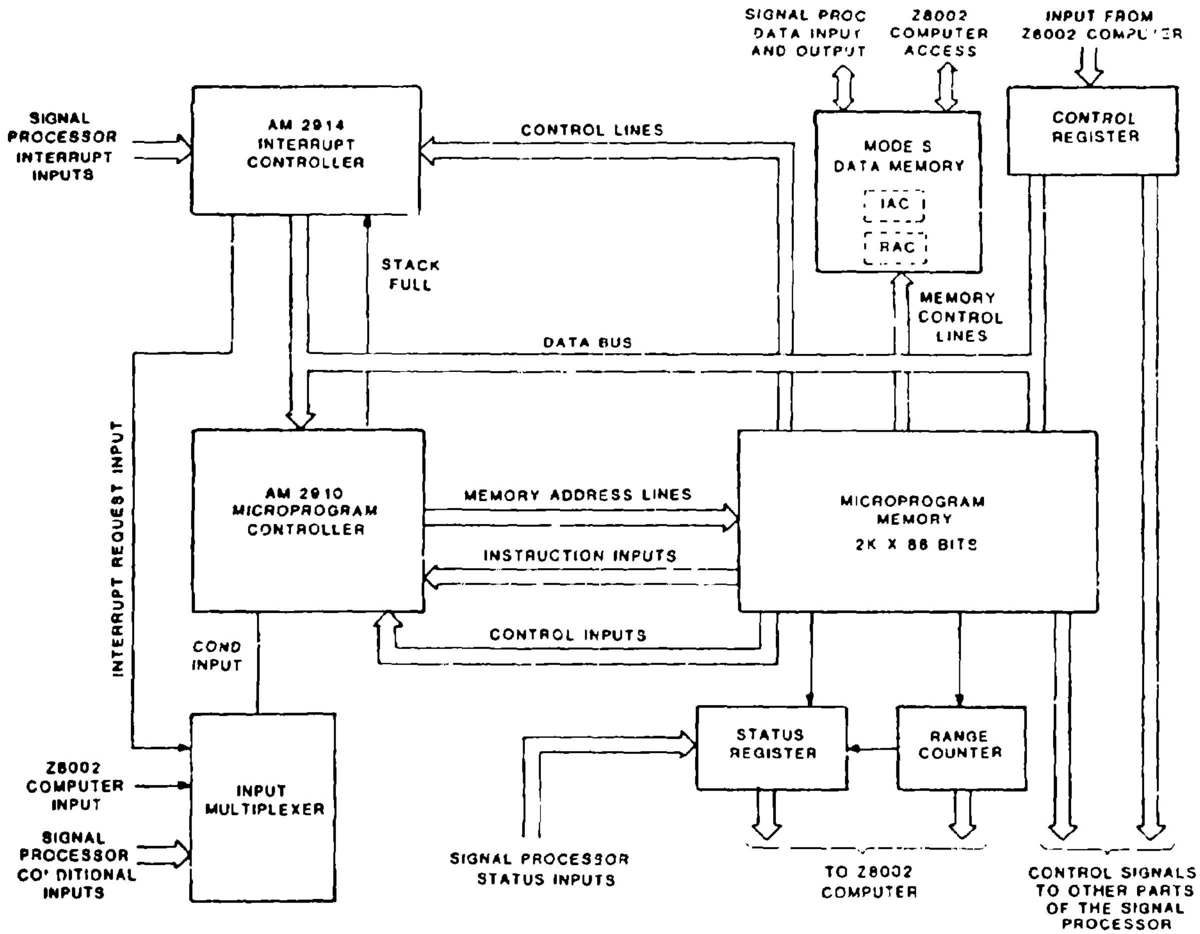


Fig. 2.5-1. Reply and interrogation controller (RIC) block diagram.

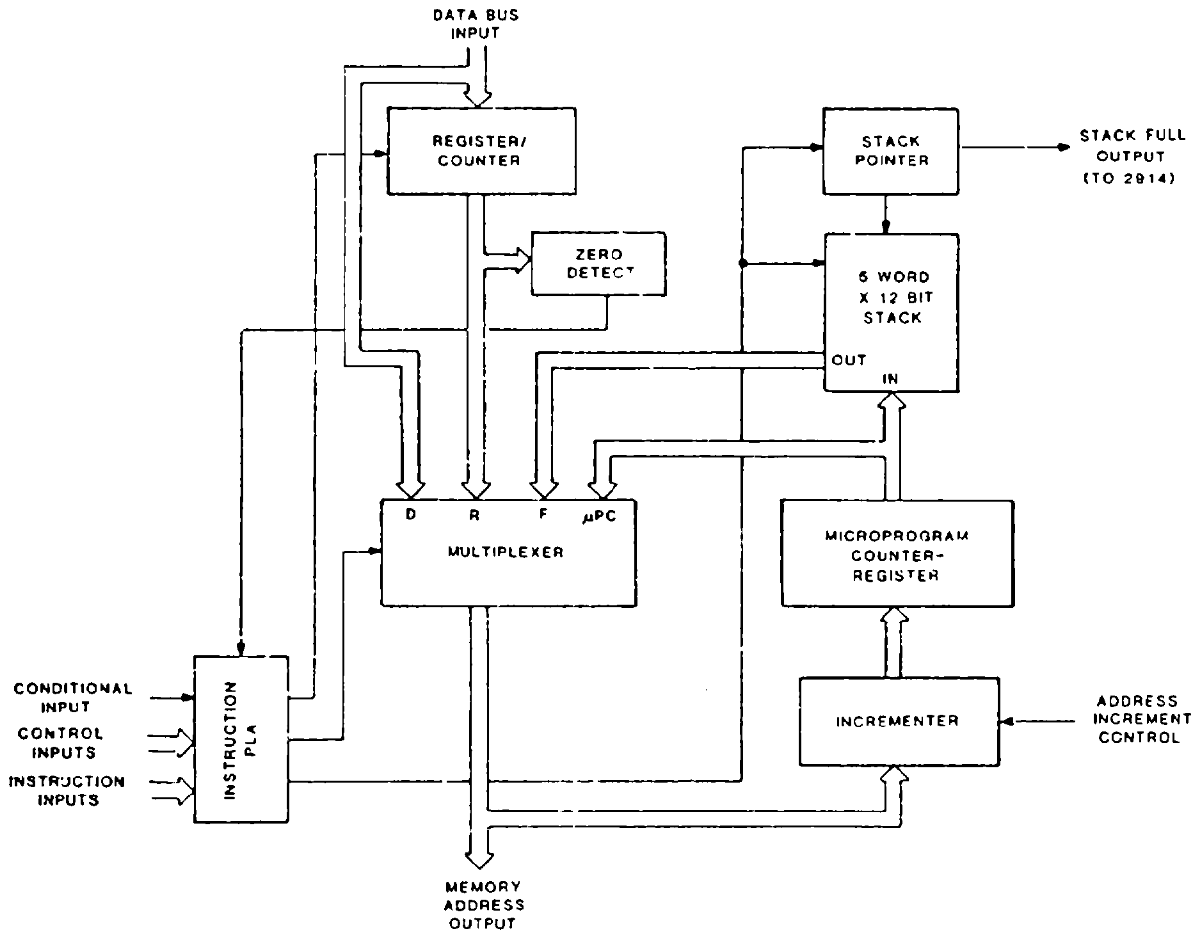


Fig. 2.5-2. Block diagram of AM2910.

is driven by a multiplexer which is used to select the condition that is to be sensed during a clock cycle. There are six different conditional inputs that can be selected during normal operation. They are:

1) Interrupt requests from the Am2914.
2) Begin present request input from the Z8002 computer. This is accomplished by the Z8002 writing to location 520 in its special I/O space.
3) ATCRBS reply accumulator full input.
4) Long or short Mode S interrogation input.
5) Long or short Mode S reply input.
6) A logical "1" input to provide for a forced pass condition.

Each of these is sensed during task execution to appropriately modify program flow.

2.5.3 Am2914 Interrupt Controller

The Am2914 is a high-speed, eight-bit priority interrupt controller that receives interrupt requests from the signal processing hardware. The interrupt inputs are recorded in the interrupt register and logically ANDed with the corresponding bits in a mask register. The result is sent to an 8-input priority encoder which produces a three-bit encoded vector representing the highest numbered active input which is not masked. An internal status register holds the value of the lowest priority interrupt which will be accepted. The output of the priority encoder is compared with the contents of the status register and if the current request is greater than or equal to the status, an interrupt request is sent to the Am2910. The Am2910 can then read the encoded vector and execute the appropriate interrupt routine.

The Am2914 is controlled by outputs from the microprogram memory. There are 16 microinstructions which allow manipulation of the mask register, status register and vector output. These are initialized for each task requiring interrupt service.

The interrupt inputs to the Am2914 were selected as a result of the need to sense more than one input at a time. Since the Am 2910 can sense on only one input at a time, the Am2914 senses those inputs that must be monitored simultaneously and relays the occurence of any one of its inputs to the Am2910 along with a vector identifying the input. The inputs defined for the Am2914 are as follows:

1) Am2910 Stack Full (Highest Priority)
2) Mode S Reply Address Counter (RAC) Overflow
3) Mode S Reply Preamble Detection
4) Mode S Squitter Preamble Detection
5) Stop Squittering Command. This is accomplished by writing to location 522 in the Z8002 special I/O space.
6) Range Counter Overflow

Any combination of these inputs can be enabled or disabled by using the masking feature of the Am2914.

2.5.4 Microprogram Memory

The microprogram memory contains all the microprogram firmware necessary to the execute the 5 predefined collision avoidance tasks and 4 diagnostic tasks. The memory is designed for a maximum size of 2K words with a word being 88 bits wide. Of the 88 bits in a word, 86 bits are defined and 2 bits are undefined and available for future use. Figure 2.5-3 shows the field definition for the 88 bit microprogram word. All 9 tasks require a total of 325 words of memory and are located in the first 625 words of the address space. The memory is built using Am27s181 bipolar PROMs which provide high speed access enabling the signal processing to run at the required speed.

2.5.5 Control Register

In order for the Z8002 computer to communicate with the RIC, a control register (see Fig. 2.5-1) is provided which the computer can load with the required control information prior to starting a particular task. The control register is 16 bits wide and is located at locations 512 and 513 in the Z8002 special I/O address space. Eleven of the bits in the register are defined and five are undefined and are available for future use. Figure 2.5-4 shows the control register field definition.

The four most significant bits (bits 13 through 16) of the control register represent the task control field which specifies what task the RIC is to perform next. Nine of the sixteen possible tasks have been defined and are listed in Fig. 2.5-5.

The 4 bit MTL control field (bits 8 through 12) allows the Z8002 computer to dynamically set the receiver sensitivity. This field controls the minimum threshold over the range -72 dBm to -40 dBm in 2 dB increments. All zeroes in the MTL field represent -72 dBm while all ones represent -40 dBm.

Three single bit fields are also contained in the control register. The first bit (bit 8) selects the antenna to be used in the present task. A 1 in this position selects the upper antenna and a 0 selects the bottom antenna.

The second bit (bit 7) sets the length of the expected Mode S reply in response to the last Mode S interrogation. A 1 in this bit indicates a long reply is expected and a 0 indicates a short reply is expected.

The last active bit (bit 6) in the control register represents the mode (Mode S or Mode C) of the current task. A 1 represents the Mode C mode and a 0 represents the Mode S mode.

The least significant five bits of the control register are undefined.

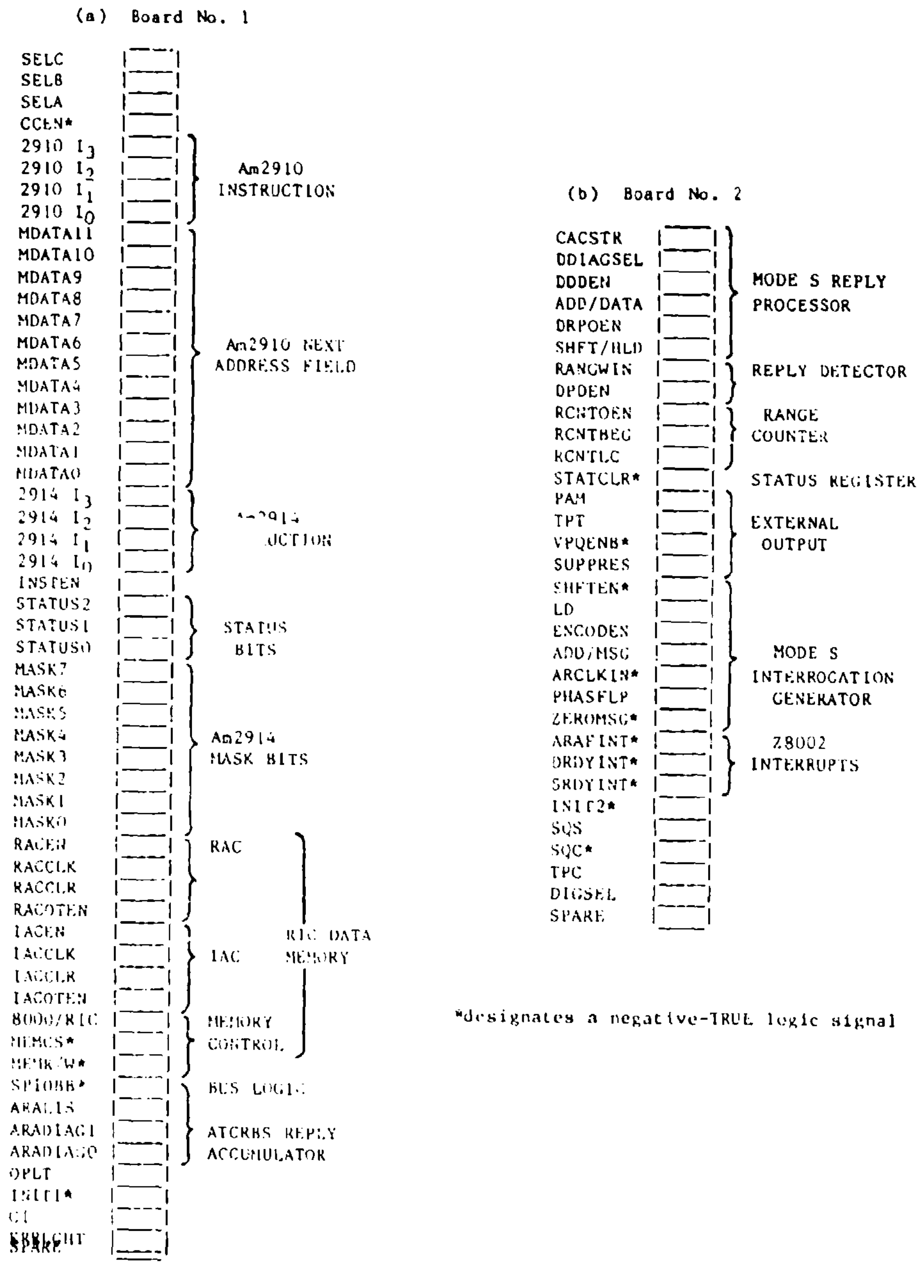


Fig. 2.5-3. Microprogram field definitions of the signal processor.

ATC-115

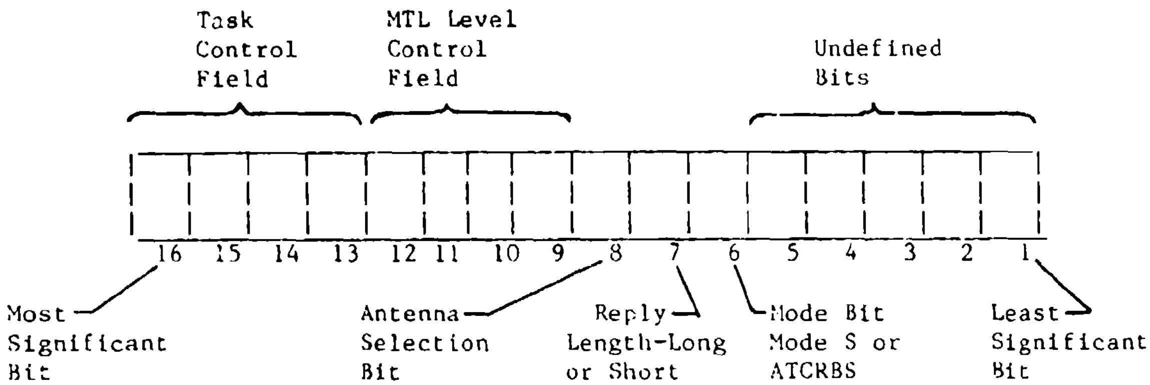


Fig. 2.5-4. RIC control register field description.

0000 - Mode C With No P4 Pulse
0001 - Mode C With P4 Pulse
0010 - Short Mode S Interrogtion
0011 - Long Mode S Interrogation
0100 - Squitter Listening
0101 - Undefined
0110 - ATCRBS Diagnostic #1
0111 - ATCRGS Diagnostic #2
1000 - Mode S Diagnostic #1
1001 - Mode S Diagnositc #2
1010 - Undefined
1011 - Undefined
1100 - Undefined
1101 - Undefined
1110 - Undefined
1111 - Undefined

Fig. 2.5-5. RIC task control field assignments.

2.5.6 Status Register

To allow the Z8002 computer to monitor RIC operation, an 8-bit status register is provided which can be accessed by reading location 516 in the Z8002 special I/O address space. The least significant four bits have been defined while the most significant four bits are undefined and are available for future expansion. Figure 2.5-6 shows the status register bit definitions. When the least significant bit is set, the reply address counter (RAC) has rolled over meaning that an excessive number of squitters or Mode S replies have been received. If the second bit is set, the range counter has reached its maximum and rolled over. When a Mode S interrogation is sent out and none of the replies received are correct, the 3rd bit in the status register is set. The fourth bit in the status register is used to indicate when the RIC is in the squitter mode. This bit is set when the squitter mode is entered and cleared when exiting the squitter mode.

2.5.7 Range Counter Latch

The RIC unit has a 10-bit range counter which is used in the Mode S mode operation. The counter is clocked at 8 MHz allowing for a maximum range window of 8.7 miles. A latch is connected to the counter which latches the range when a preamble is detected. The latch can then be accessed by reading locations 518 and 519 in the Z8002 special I/O space. When the mixímum range is reached the counter halts and a bit in the status register is set.

2.5.8 Mode S Data Memory

A Mode S data memory is included in the RIC consisting of 512 bytes of storage, a 7 bit reply address counter (RAC), and a 7 bit interrogation address counter (IAC) (Fig. 2.5-1). The memory is accessible to both the Z8002 computer (using the special I/O instructions) and the Am 2910. The special I/O address space assignment is given in Fig. 2.5-7. The Mode S memory is also controllable from the microprogram memory by the Am 2910. The Am 2910 stores Mode S replies in the first 256 bytes of the data memory using the RAC and reads the interrogation data from the last 256 bytes of the data memory using the IAC. The Am 2910 controls who can read from the memory at any instant. The memory is normally accessable by the Z8002 except when the Am 2910 requires memory access. When the Am 2916 requires access to the memory, it locks out the Z8002. An interrupt is sent to the Z8002 if it attempts access while the Am 2910 is using the memory. The Z8002 can also read the current value of the RAC by accessing location 514 in the special I/O address space. For additional details, refer to section 2.9.4.

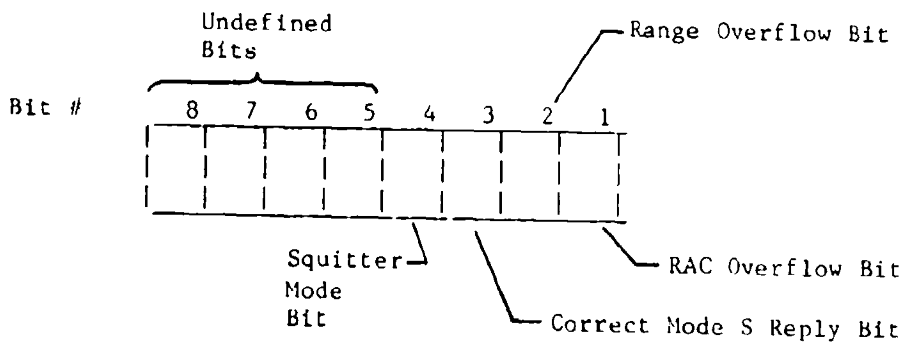


Fig. 2.5-6. Status bit definitions.

SPECIAL I/O BYTE ADDRESS (DECIMAL)	FUNCTIONAL ASSIGNMENT
0-225	Mode S Reply Area
256-511	Mode S Interrogation Area
512-513	Control Register (16 Bits)
514	Reply Address Counter (8 Bits)
516	Status Register (8 Bits)
518-519	Range Counter (16 Bits)
520	Begin Present Request Command
522	Stop Squitter Listening Command

Fig. 2.5-7. Special I/O address assignments.

2.6 Mode S Interrogation Generator (DIG)

2.6.1 General Description

The Mode S interrogation hardware is used to perform functions of encoding the Mode S data and properly modulating the transmitter. This activity is controlled and coordinated by the RIC. When a Mode S interrogation is required, the Z8002 loads the 56 or 112 bits of interrogation data into the Mode S memory starting at location 256. The RIC then loads this data, sixteen bits at a time, into the DIG as it is encoded and sent. The last 24 bits of a Mode S message contains the address of the Mode S target and is stored in the DIG for comparison with reply addresses received by the Mode S reply processor.

2.6.2 Mode S DPSK Encoder

Mode S interrogations are encoded using a 24-degree generator polynominal [2,3]. The feed-out encoder circuit used in the DIG is shown in Fig. 2.6-1. A message is encoded by shifting the first 32 bits (88 bits for a long message) into the input with gates G-1 and G-4 closed and gates G-2 and G-3 open. After the 32nd (or 88th) bit has been input, the content of the register is the partial quotient resulting from division by the generator polynominal. If the encoder were shifted 24 more times with just G-3 enabled, the appropriate parity bits to overlay on the encoded address would be generated at the output. To complete the encoding process, gates G-1 and G-4 must be opened and gates G-2 and G-3 closed so address bits are multiplied by the generator polynominal and simultaneously overlayed with the parity bits of the first 32 bits (88 bits for long message).

2.7 Mode S Reply Processor (DRP)

2.7.1 General Description

The Mode S reply hardware is used to decode Mode S replies which are received as a result of a Mode S interrogations and Mode S squitters which occur asynchronously. The Mode S reply data block is encoded using pulse position modulation (PPM) where each bit is represented by 1.0 sec interval. A 0.5 sec pulse is transmitted in the first half of the interval if the data bit is a 1, and in the second half of the interval if the data bit is a 0. The VPQ generates a digital signal called chip amplitude compare (CAC) which contains the pulse position information from the reply. The DRP then samples the CAC signal under the synchronized control of the RIC to generate the 56 or 112 bit received serial bit stream. The serial bit stream is decoded using a feed-out decoder similar to that used in the Mode S Interrogation Generator (DIG). As the data is decoded, it is stored 16 bits at a time in the Mode S data memory beginning at location zero. In the case of a Mode S reply, as the last 24 bits are decoded, they are also serially compared with the address which was stored in the DIG when the interrogation was sent. The result of the comparison is stored in the status register.

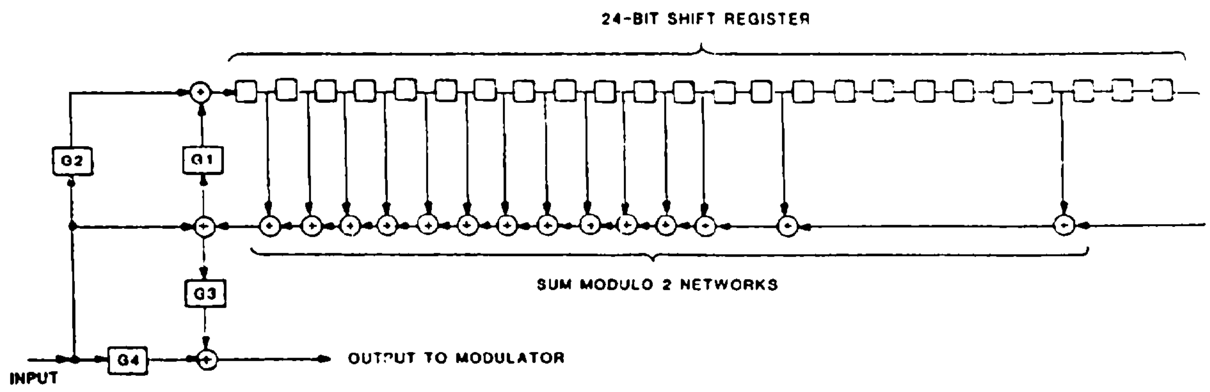


Fig. 2.6-1. Mode S interrogation generator feed-out encoder.

ATC-117

2.7.2 Mode S Downlink Decoder

Mode S replies are encoded using the same 24-degree generator polynominal [2,3]. The feed-out decoder used is shown in Fig. 2.7-1. When a message is received, the Mode S reply preamble is detected by the reply decoder and the RIC is interrupted to start message processing. The decoder is configured with gates G-1 and G-2 closed and G-3 open while the first 32 bits (88 bits for a long message) are received. Following the 32nd bit (or 88th bit), gates G-1 and G-2 are opened and G-3 is closed while the last 24 bits of message are decoded. If the reply received was from the correct aircraft and no errors occurred during reception and decoding, the last 24 bits received are the address of the aircraft interrogated by the DIG.

2.8 Mode C Reply Accumulator

2.8.1 General Description

Mode C replies are received and stored in digitized pulse form in the Mode C reply accumulator (CRA), Fig. 2.8-1. The CRA contains an address counter, 1024x2 bit memory, control logic and a logic circuit which does a logical OR between three consecutive data samples in the leading edge data stream coming from the reply detector. The CRA is controlled by the RIC which selects who has access to the RIC bus. The CRA is selected by the RIC for loading from the reply detector only after an Mode C interrogation has been sent. Following a Mode C interrogation, the RIC pauses to account for the transponder turn around delay and then activates the CRA logic. The logic locks out any accesses from the RIC bus and begins generating the timing signals to load data into the CRA memory. The clock speed of the memory and the reply detector is 8.27 MHz. After the memory is full, the RIC reconnects the CRA memory to the RIC bus and interrupts the Z8002 to let it know that new data is available in the RIC.

2.8.2 CRA Storage Memory

The storage memory in the CRA is configured as a 1024x2 bit memory where the two bits are connected to D1 and D0 and the RIC bus. D0 is the sum of all leading edge data pulses (ΣLE) as received from the reply detector. D1 is the logical OR of three consecutive ΣLE data bits. Thus, when the CRA logic is activated, the memory is cycled sequentially beginning at location zero. The current logical value of the ΣLE and the Logical OR of the previous, present, and following ΣLE values is stored as each consecutive location is addressed. This storage form was used to simplify the Mode C software reply processing which requires the OR of the consecutive ΣLE samples. In addition, by storing the data sequentially as it is received, the CRA address space correlates directly with the range of each target found in the memory during Mode C reply processing.

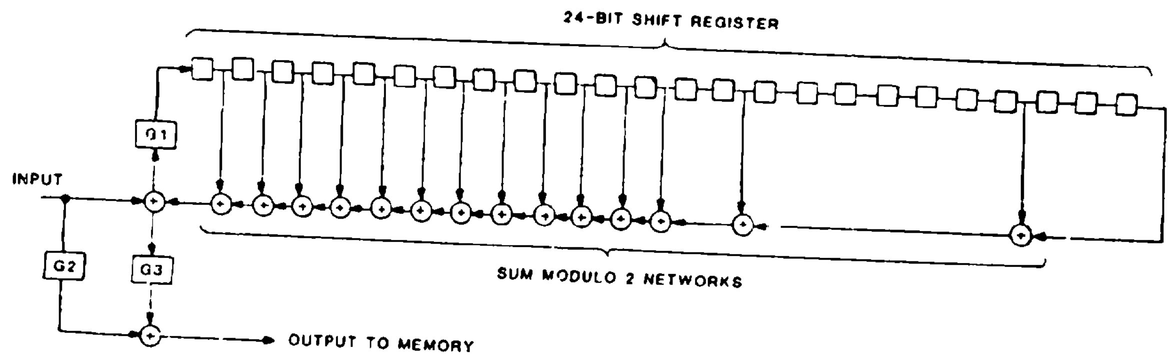


Fig. 2.7-1. Mode S reply processor feed-out decoder.

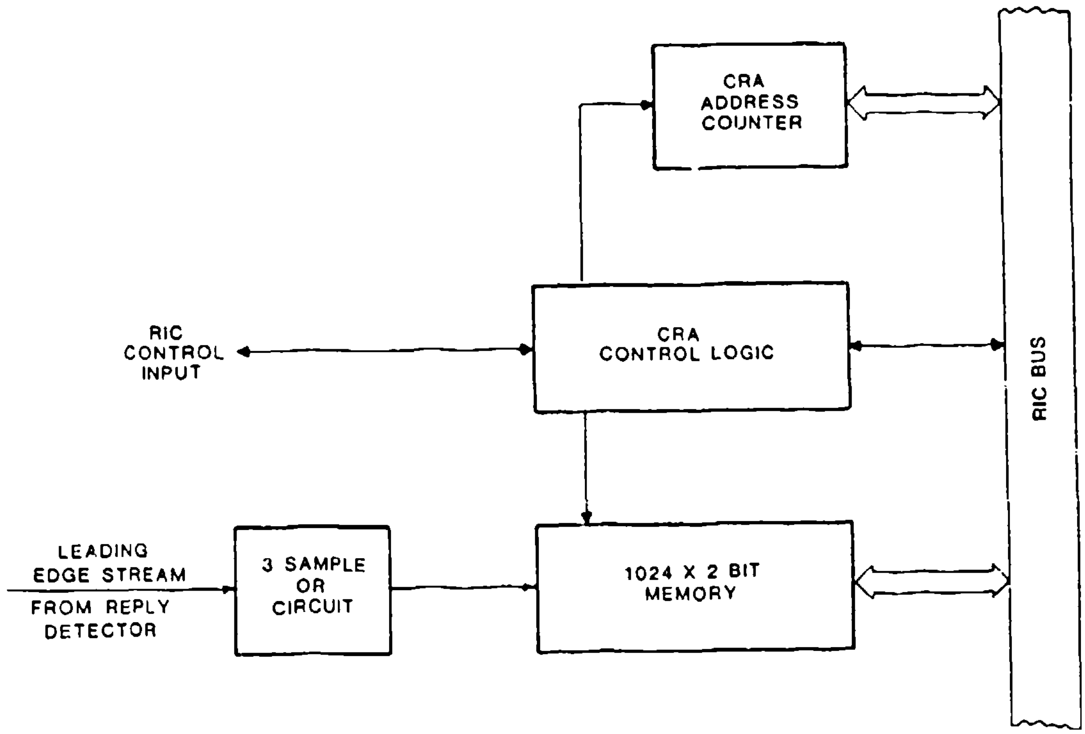


Fig. 2.8-1. Mode C reply accumulator (ARA).

2.9 Computer Subsystem

2.9.1 General Description

The computer in GATCAS is based on the AmZ8002 microprocessor. It was selected to perform reply and interrogation processing as well as the tracker and collision avoidance algorithms in a stand-alone unit. However, as discussed in 2.1.1, the GATCAS was built to work with an air-carrier TCAS unit and therefore only executes reply and interrogation processing tasks. Figure 2.9-1 shows a block diagram of the computer subsystem. A single board computer (Am96/4016) and a memory I/O board constitute the computer subsystem. The Am96/4016 has two serial I/O ports, a counter timer, and one parallel I/O port. The serial ports are designated for use in data recording and displaying data to the pilots in the stand-alone unit. However, in the design implemented, one port is used to communicate to the air-carrier unit and the other is not used. The parallel port is used to communicate and download programs from the AmSYS 8/8 software development system. The counter-timer is used to generate the baud rate for the serial I/O ports.

The memory I/O board includes 31 kilobytes of random access memory (RAM), 160K bytes of programable read only memory (PROM), two serial I/O ports, one parallel I/O port, one system timing controller containing five general purpose 16-bit counters, and two cascaded interrupt controllers. One serial I/O is used to drive a console device and the other is a spare. The parallel port is divided into two 8-bit ports and two 4-bit ports. In the stand-alone unit, the 8-bit ports are used together to input the 12-bit encoded altitude for the aircraft and one of the 4-bit ports is used to input the pilot sensitivity level. Neither of these functions is used in the unit presently implemented. One 4-bit port is used to output a page address to the PROM memory. Vectored interrupts are used to notify the Z8002 of time-critical events. The two cascaded interrupt controllers are each capable of eight requests providing a total of 16 prioritized vectored interrupts. The system timing controller provides a time-of day clock and programmable time delays used to generate time-based interrupts.

2.9.2 Z8002 Microprocessor

The AmZ8002 is a 16-bit microprocessor which can directly address 64 kilobytes of memory. Its architecture embodies sixteen 16-bit general purpose registers. Facilities are provided to maintain three distinct memory address spaces - code, data, and stack, as well as two separate I/O spaces - normal and special. The Z8002 implements a powerful instruction set including 110 instruction types, eight addressing modes, auto-indexing instructions, and string instructions with repeat and non-repeat versions. These instructions operate on several data types: bit, byte (8-bits), word (16-bits), long word (32-bits), byte string, and word string. The system can execute instructions in one of two modes - system and normal. For further details concerning the AmZ8002, refer to Reference 1.

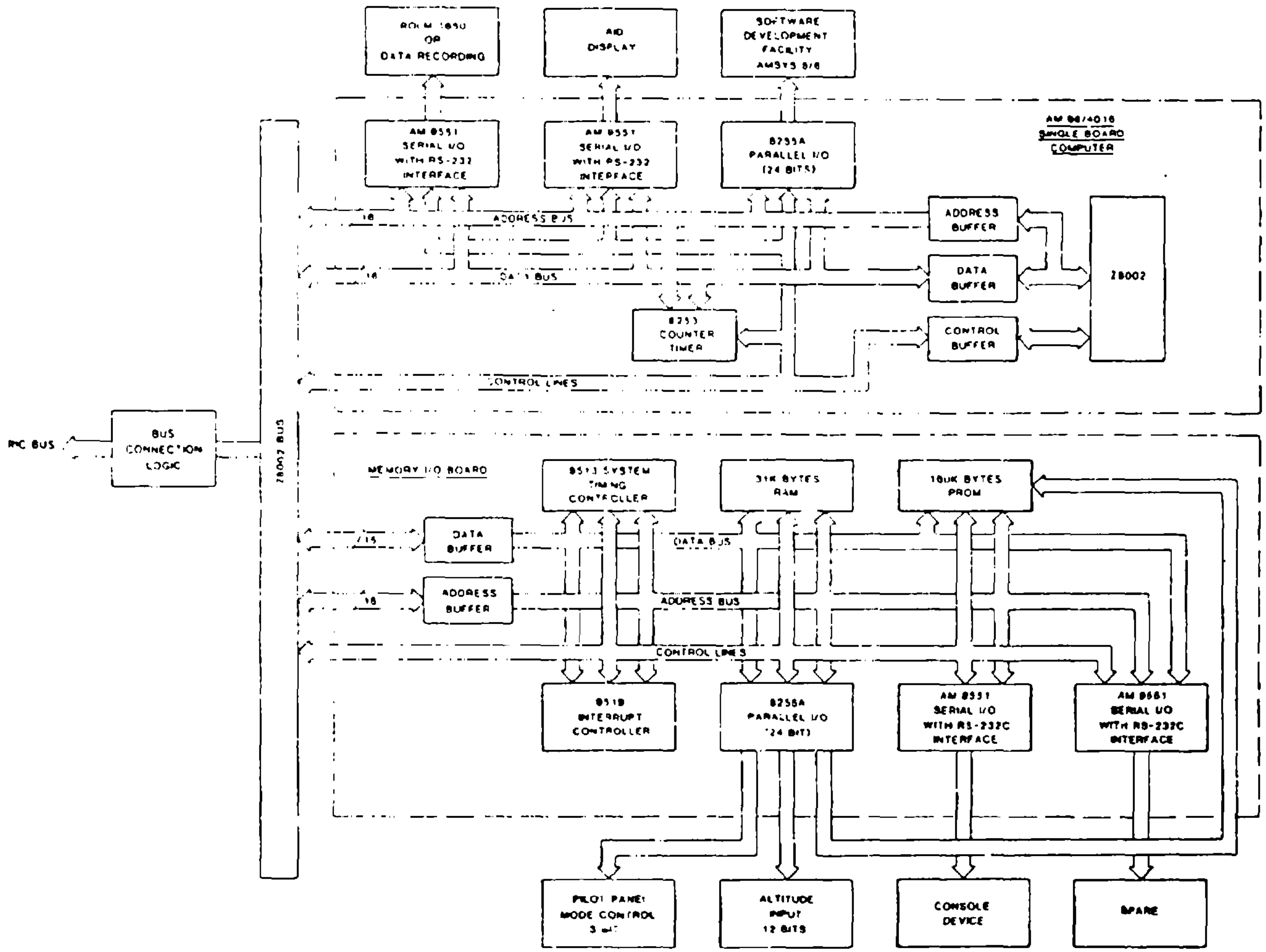


Fig. 2.9-1. Computer subsystem.

2.9.3 System Memory

The entire memory used by the Z8002 is located on the memory I/O board. Since the Z8002 is only capable of directly addressing 64 kilobytes of memory and more memory than this was required, additional memory was added through a paging scheme utilizing 4 bits from the parallel port on the memory I/O card (Fig. 2.9-2). The first 16 kilobytes (addresses 0000 to 3FFF HEX) is PROM which is accessible at all times. The next 16 kilobytes (addresses 4000 to 7FFF) is a PROM page which can be selected through the parallel port to be one of nine independent pages. Each page is selected when the code on that page needs to be accessed. The uppermost 32 kilobytes (addresses 8000 to FFFF HEX) of memory are RAM and can be accessed continuously; 8000 to 83FF is assigned to the CRA and is read only. Thus the total useable memory is 31 kilobytes of RAM and 160 kilobytes of PROM.

2.9.4 I/O Address Space

The Z8002 has two I/O address spaces in addition to the 64K memory space. This is possible by decoding the address together with the four status lines from the CPU which indicate the nature of the current transaction. The Z8002 has two sets of I/O instruction (normal and special) which go with each respective I/O space. The serial ports, parallel ports, system timing controller, counter timer, and interrupt controllers are assigned to normal I/O space. Figure 2.9-3 contains a listing of the address assignments for the normal I/O addressing space. The least significant 12-bits have been decoded giving a total of up to 4K I/O devices.

The Reply and Interrogation Controller (RIC) memory and ports which must be accessed by the Z8002 have been assigned to the special I/O space. Figure 2.5-7 shows the special I/O addressing space assignment. The least significant 10-bits of the address have been decoded giving a total of 1024 bytes in the RIC accessible from the Z8002. The two locations assigned to Begin Present Request and Stop Squitter Listening are used by the Z8002 to start and stop the RIC. By writing to these locations, a request can be started or squitter mode processing can be halted. For additional details, refer to section 2.5.8.

2.9.5 Am8255A Parallel I/O Port

The Am8255A is a general purpose programmable parallel I/O device which is used on both the memory I/O board and the Am86/4016. It has 24 bits which can be programmed in two groups of twelve, utilizing three modes of operation. In the first mode, each group of twelve bits may be programmed in sets of 4 and 8 to be input or output. In the second mode, each 12 bit group may be programmed to have 8 lines of input or output and 3 of the remaining four pins for hand shaking and interrupt control signals. The last mode, is a bi-directional bus mode which uses eight lines for a bi-directional bus, and five lines (borrowing one from the other group) for hand shaking. For additional details about the 8255A, see references 3 and 4.

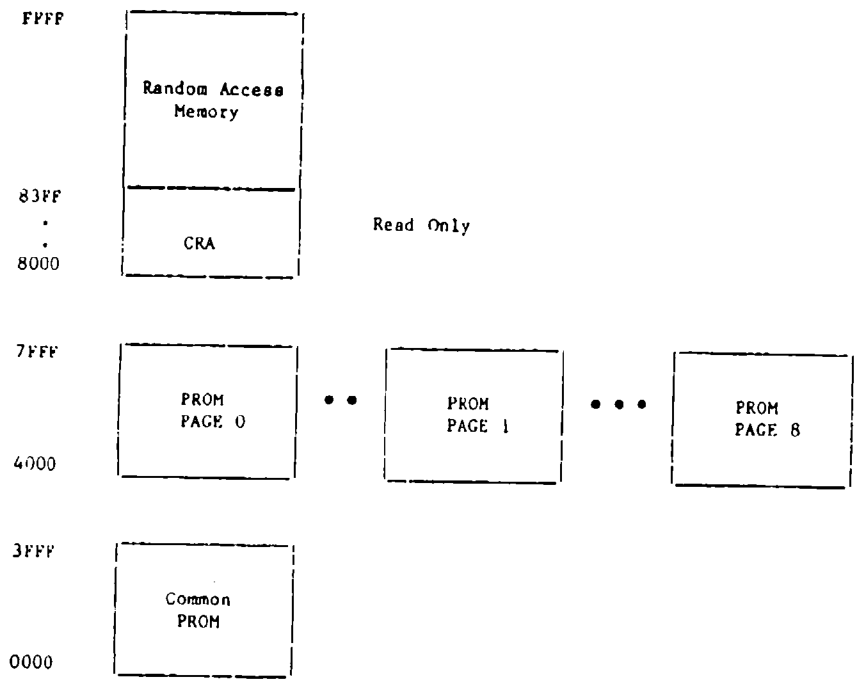


Fig. 2.9-2. Z8002 program memory address assignment.

NORMAL I/O ADDRESSING SPACE

```
FFC-FFF : N/A

FFB-FF8 : N/A

FF4-FF7 : N/A

FF3-FF0 : Am8255A=Parallel port #1
    FF3 : Control port
    FF2 : Port C
    FF1 : Port B
    FF0 : Port A

FEF-FEC : Am9551 RS232 port #1
    FEF : Control
    FEE : Data
    FED : Control (same as FEF)
    FEC : Data (same as FEE)

FEB-FE8 : Am9551 RS232 port #2
    FEB : Control
    FEA : DAta
    FE9 : Control (same as FEB)
    FE8 : Data (same as FEA)

FE7-FE4 : Am8253 System Timing Control #1
    FE7 : Mode control
    FE6 : Counter 2
    FE5 : Counter 1
    FE4 : Counter 0

FE3-FE0 : Keyboard return lines

FDR-FDC : Keyboard scan lines

FD8-FDB : Single-step control

FD7-FD4 : Breakpoint register

FD3-FD0 : LED display
    FD3 : 2-th character (left most)
    FD2 : 19th character
    FD1 : 18th character
    FD0 : 17th character

FCF-FCC
    FCF : 16th character
    FCE : 15th character
    FCD : 14th character
    FCC : 13th character
```

Fig. 2.9-3. I/O address space assignment.

NORMAL I/O ADDRESSING SPACE

```
FCB-FC8 :
    FCB : 12th character
    FCA : 11th character
    FC9 : 10th character
    FC8 :  9th character

FC7-FC4
    FC7 :  8th character
    FC6 :  7th character
    FC5 :  6th character
    FC4 :  5th character

FC3-FC0
    FC3 :  4th character
    FC2 :  3rd character
    FC1 :  2nd character
    FC0 :  1st character (right most)

FBF-FBC : 8255A Parallel port #2
    FBF : Control post
    FBE : Port C
    FBD : Port B
    FBC : Port A

FBB-FB8 : Am9551 RS232 port #3
    FBB : Control
    FBA : Data
    FB9 : Control (same as FBB)
    FB8 : Data (same as FBA)

FB7-FB4 : Am9551 RS232 port #4
    FB7 : Control
    FB6 : Data
    FB5 : Control (same as FB7)
    FB4 : Data (same as FB6)

FB3-FB0 : Am9513 system timing controller #2
    FB3 : Control or Status
    FB2 : Control (same as FB3)
    FB0 : Data (same as FB2)

FAC-FAF : Am9519A Universal Interrupt Controller #1
    FAF : Controller status
    FAE : Data
    FAD : Control (same as FAF)
    FAC : Data (same as FAF)

FAB-FA8 : Am9519A Universal Interrupt Controller #2
    FAB : Controller
    FAA : Data
    FA9 : Control (same as FAB)
    FA8 : Data (same as FAA)
```

Fig. 2.9-3. I/O address space assignment (cont'd).

2.9.6 Am9551 Serial I/O Port

The Am9551 is a programmable serial data communication interface located on the memory I/O board and the Am96/4016 board that provides a universal synchronous/asynchronous receiver/transmitter (USART) function. It is normally used as a peripheral device which can be programmed to operate in several different serial communication formats. The device accepts parallel data from the CPU, formats and serializes the information based on its current operating mode, and then transmits the data as a serial bit stream. The USART can operate in an independent full duplex mode allowing simultaneous reception and conversion from serial to parallel form. Control, operation, and format options are all selected by the controlling processor. For further information about the Am9551 refer to references 3 and 4.

2.9.7 Am9519A Interrupt Controller

The Am9519A Universal Interrupt Controller is a processor support circuit located on the memory I/O board which provides an interrupt structure to improve system efficiency and versatility. A single 9519A can handle eight maskable interrupt request inputs, resolve priorities, and supply up to four byte-programmable responses for each interrupt. For applications requiring more interrupt inputs, multiple units can be cascaded to the required number.

When the Am9519A receives an unmasked interrupt request, it issues a group interrupt output to the controlling processor. When an interrupt is acknowledged, the controller outputs the one-to-four byte response associated with the highest priority unmasked interrupt request. The controlling processor can also set interrupt requests under software control. This allows for hardware prioritization of software tasks and aids system diagnostic and maintenance procedures. For additional information on the Am9519, refer to Reference 4.

2.9.8 Am9513 System Timing Controller

The Am9513 system timing controller (STC) is a circuit on the memory I/O board which is designed to service many types of sequencing and timing applications. The STC contains five general-purpose 16-bit counters which can be programmed to count up or down in either binary or BCD. Each counter can use internal or external frequency sources and provides three-state outputs which can be either pulses or levels. The accumulated count in each counter may be read without disturbing the counting process. Any of the counters may be internally concatenated to form an effective counter length of up to 80 bits. For additional information on the Am9513, refer to reference 4.

2.9.9 Am8253 Counter Timer

The Am8253 is a programmable interval timer located on the AM96/4016 computer board. This circuit can be used as a variable time delay under software control, programmable rate generator, event counter or a real time clock. In the system described here, this counter was used to generate the baud rate for the two serial ports on the AM96/4016 board. For further details about the Am8253, refer to Reference 4.

2.9.10 Bus Connection Logic

The bus connection logic provides a means for the Z8002 to give the Signal Processor control information, and retrieve data that has been collected. It is designed so that if the Z8002 and the Reply and Interrogation Controller (RIC) simultaneously access a common resource, the Z8002 is locked out and an interrupt to the Z8002 generated so that a recovery can be accomplished.

There are five parts of the signal processor which can be accessed by the Z8002: the reply address counter (RAC), control register, range counter, Mode S data memory, and the Mode C Reply Accumulator (CRA). The RA, control register, range counter, and Mode S data memory all are assigned to the special I/O space and are all connected to a bus within the RIC. The CRA is assigned locations in the normal memory space.

When the Am2910 is accessing any of the parts of the RIC, the Z8002 is locked out from further access. If the Z8002 attempts an access, an interrupt is generated to the Z8002. If the Z8002 is accessing any of the resources in the RIC, and the RIC accesses any of the common resources, the Z8002 is again locked out and an interrupt generated.

The CRA can be accessed by the Z8002 while the AM2910 is accessing any of the resources in the special I/O space. However, if the AM2910 is accessing the CRA, the Z8002 is locked out from accessing the CRA and the resources in the special I/O space. Whenever the Z8002 attempts an access which is blocked, an interrupt is generated which allows the Z8002 to take appropriate action.

3.0 GATCAS SOFTWARE

3.1 Overview

The GATCAS software consists of Z8002 microprocessor software and Am2910 microprogram software. The microprocessor software consists of a real-time executive providing a resource-sharing environment ideal for multiple real-time activities. The functions performed by the microprocessor software are: communications with the TEU, execution of Mode S and Mode C interrogation commands from the TEU, Mode S and Mode C reply processing; and diagnostic testing. A description of the microprocessor software is given in section 3.2.

The microprogram software is an event-driven program that commands Mode C and Mode S interrogation and processes replies under the control of the microprocessor software. It also performs diagnostic functions. Section 3.3 describes the microprogram software.

3.2 Real-Time Microprocessor Software

The GATCAS microprocessor software system configured to utilize the TEU is a real-time executive patterned after the DEC* RSX-11S. The RSX-11 family of operating systems is designed to provide resource-sharing among multiple real-time activities. The basic program unit which the operating system services is called a task. Tasks are scheduled to execute on an event-driven basis by a task scheduler. Input/output (I/O) is interrupt driven and is accomplished by I/O handlers. Event flags associated with significant events are used by tasks to achieve efficient synchronization between themselves and other software tasks. A task can set, clear, test, and wait for any event flag as well as change its own priority, receive or send I/O messages, or ask to be awoken after a specified time. These activities are accomplished by a task through the use of task directives.

Three main tasks run during normal operation: the REQUEST task, the Mode C reply processing task, and the OUTPUT task. The REQUEST task handles RIC requests and handles processor input buffers from the I/O handlers. The Mode C reply processing task processes the Mode C replies collected by the Mode C Reply Accumulator. The OUTPUT task formats and sends data to the TEU.

3.2.1 Task Scheduler and I/O Handler

The GATCAS unit contains one microprocessor (the central processing unit or CPU) and executes one task at a time. Thus, the tasks within the system must share the CPU in order that all tasks can execute within the one-second cycle period of the GATCAS. This is accomplished by the use of a task scheduler (executive) which continually monitors tasks in the system allowing the highest priority executable task to run.

Tasks, in general, depend upon input and information from other tasks and system routines to execute. The availability of the input or information is communicated between tasks through the use of event flags. As a task executes

*Digital Equipment Corporation.

and needs input, it executes the task directive WAITF. This directive will test a specified flag and if its state is false (or blocking), the task scheduler halts its execution and starts executing the highest priority task that is not blocking. If the flag is true, the current task will continue to execute. A task that has halted because a flag was blocking can continue execution when the flag becomes true (or unblocking) if it is the highest priority unblocked task in the system. The task scheduler continually transfers control of the CPU to the highest priority executable (unblocked) task.

Interrupt driven I/O handlers are not controlled by the task scheduler. Unmasked interrupts associated with significant events automatically interrupt task execution in hardware and allow I/O routines to perform their function. Interrupt routines that do not alter system variables execute and return control either to the task running prior to the interrupt or the task scheduler. Control is returned to the task scheduler when rescheduling, based on the action performed by the I/O routine, is necessary.

3.2.2 Task Directives

Task directives are issued by tasks to perform specified functions. The following section describes the directives available in the GATCAS unit.

3.2.2.1 Create Task (RTSK)

Parameters:

- ASCII name of the task
- entry address, including page addressing
- priority of task
- how many data blocks a task will use
- size of stack needed by task

This directive sets up and initializes a task control block, data blocks and stack space.

3.2.2.2 Mark Wait (MARKWT)

Parameters:

- amount of time (30 bit word, lsb = 0.25 microsec)
- which data blocks to use
- which flag to use

The task issueing this directive is suspended for the time specified.

3.2.2.3 Mark Return (MARKRT)

This directive makes the flag blocking and requests that it be unblocked after the specified amount of time. The issuing task is not suspended.

3.2.2.4 Update Time (UPDTM)

This directive requests that system time, STO, (30 bit word, lsb = 0.25 microsec) be brought up-to-date.

3.2.2.5 Wait for Flag (WAITF)

Parameter:

- name of flag

This directive is used by a task to synchronize itself with other tasks, I/O drivers, or the RIC.

When this directive is issued there are two possibilities:

- the flag was blocked, in which case the task is suspended, or
- the flag was unblocked, in which case the task just continues to run, and the flag reverts to its blocking state.

When the scheduler unblocks a task, the flag that caused this action is simultaneously put into the blocking state. A feature of the Z8002 instruction set is that a flag can be tested and reset to the blocking state by a single indivisible instruction, which simplifies the process of maintaining the integrity of the synchronization process.

3.2.2.6 Change Priority (CBGPRI)

Parameter:

- new priority

This directive allows a task to raise or lower its priority.

3.2.2.7 Transmit (XMITR)

Parameters:

- port number
- data block number
- buffer

This directive requests the RS232C I/O driver to transmit the contents of the buffer to the specified port. The request is queued and control returns to the issuing task.

The first byte of the buffer header is the flag that will be made unblocking by the I/O driver when the transmission is complete. This allows a WAITF(buffer) to synchronize the task with the I/O. The number of bytes to transmit is contained in the buffer header.

3.2.2.8 Receiver (RCVR)

Parameters:

- port number
- data block number
- buffer
- size of buffer in bytes

- matching byte
- binary/ASCII
- time-out count (lsb = 16.38 millisecs)

Input requests from the air-carrier TCAS are received in binary mode with matching byte an ASCII 'R'. In binary mode the second byte contains the number of 16 bit words that follow. The time-out counter starts running when the matching byte has been found.

Console input is received in ASCII mode, the matching byte being 'R' for input destined to the REQUEST task, 'C' for the CONTROL task and 'T' for the TEST task. Since the input is ASCII, it is terminated by the RETURN key.

The data in either the ASCII mode or binary mode is put into a buffer, and the number of data bytes is put in the buffer header. The matching byte is not part of the data. In general the I/O driver has several requests for input queued to the same port, those with duplicate matching bytes imply double buffering is being used. When the I/O driver has a complete message in a buffer, it unblocks the buffer and asks for tasks to be rescheduled, the result being that the highest priority unblocked task runs.

3.2.2.9 Flush Buffer (FLUSH)

Parameters:

- port number
- matching byte

This directive searches the list of input requests queued to the I/O driver for the specified port. All requests that have the specified matching byte are marked as timed out, and the buffer unblocked. When the task which issued the input request executes, it can check the buffer header for time-out and reject the input message. This flushes out old requests so that new ones can be issued for a different port. This mechanism is used by the CONTROL task to switch the REQUEST task to local console input.

3.2.2.10 Simulate Input (SIMIN)

Parameters:

- port number
- buffer
- matching byte

This directive asks the I/O driver to treat the contents of the specified buffer similarly to a byte stream coming in the specified port with the given matching byte as the first character of the stream. This is used by the TEST task to generate low-rate, repetitive, input requests.

3.2.3 Mode C Processing

The ATCRBS reply processing in the GATCAS unit is unlike that done in the TEU in that all processing, after the received signal is digitized, is done in software. The ATCRBS reply waveform consists of framing pulses spaced

20.3 microseconds apart, with information pulses spaced every 1.45 microseconds between the framing pulses, and a special position identification pulse spaced 4.35 microseconds following the last framing pulse. As the replies are received, they are digitized at an 8.27 MHz rate. This results in 12 samples per information pulse position. From this sample data stream, the video pulse quantizer (Section 2.3) generates a LE data stream which is stored in memory in an array called SMP(K). The array is filled with values of 1 or 0 where a 1 indicates the presence of a leading edge and 0 marks the absence of a leading edge. SMP(K) is copied into the Z8000 system memory where it is analyzed to determine the range and altitude of responding targets.

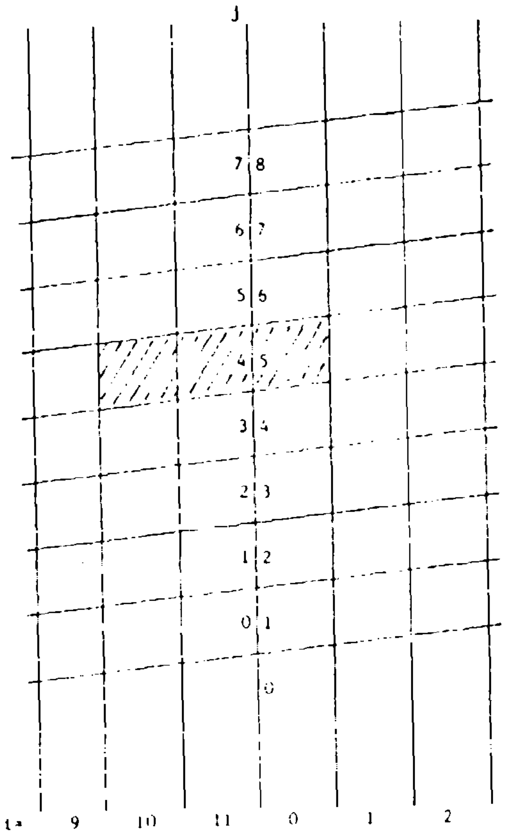
For ease of visualizing the ATCRBS processing algorithm, the array SMP(K) could be viewed as being arranged on a helix whose pitch is 1 in 12. Thus each sample lies along a line parallel to the axis of the helix as shown in Fig. 3.2-1. Consider the two dimensional array H(i,j) where

$$H(i,j) = SMP(i+12*j) \text{ for } 0<i<11 \text{ and } 0<j<71.$$

An ATCRBS reply, R(i,j) is provisionally declared if H(i,j)=1 and N1(H(i,j+14))=1 where N1 is a "neighborhood function" which equals the value of the logical OR of H(i,j+13), H(i,j+14), and H(i,j+15). This value is generated and stored by the RIC in a secondary array provided with SMP(K). It need not be calculated in software but merely accessed from that secondary array. As each reply is found, it is associated with one of 12 bits according to the i subscript.

When a new reply is found, its list (the primary list) and the two adjacent lists are examined to see if they contain overlapping replies. Each reply can be classified in one of three possible ways: NORMAL, POSSIBLE-PHANTOM, or PHANTOM. When the three lists are checked, the classification of the new reply is determined and the classification of the overlapping replies in each list is appropriately updated.

The primary list is processed first followed by the adjacent lists. The new reply initially has a default classification of NORMAL. If the first bracket pulse of the new reply is found to lie in the C2 information pulse position of an overlapping reply, it is reclassified as PHANTOM and all processing stops. Otherwise, processing continues and the overlapping replies are checked to see if a NORMAL reply overlaps the new reply. If so, the new reply is reclassified as POSSIBLE-PHANTOM. If an overlapping reply is classified as POSSIBLE-PHANTOM, it is reclassified as PHANTOM. When the primary and adjacent lists have been completely processed, the adjacent lists that are a distance of two from the primary list are checked for overlapped replies marked POSSIBLE-PHANTOM. If any are found, they are reclassified as PHANTOM. The last phase of the ATCRBS processing determines the garble bits in valid replies due to reply overlap and flags these bits in each reply. All PHANTOM replies are ignored. The range of valid targets is calculated from the reply position in memory, and altitude is determined from the information pulses within the brackets. For further information about ATCRBS processing, refer to references 5 and 6.



NI neighborhood (11,4) is the logical OR of (10,4), (11,4), (0,5)

Fig. 3.2-1. Helical representation of ATCRBS Leading Edge (LE) pulse stream.

3.2.4 Mode S Processing

When requested to interrogate a Mode S target, the RIC reads, encodes, and modulates the transmitter with the Mode S data stored in the data memory by the Z8002. If a target responds, the reply is received, decoded, and stored in the Mode S data memory. Then the RIC interrupts the Z8002, and the Mode S data, if any, is copied into the Z8002 main memory. The Z8002 can test to see if a reply was received and, if so, whether it was correctly decoded. Mode S replies require no further processing except to be properly formatted for transfer to the TEU. This is done by the output task when the data is sent to the TEU.

3.2.5 Squitter Processing

The RIC is normally in the squitter listening mode for more than 99% of the one second cycle in which the GATCAS unit performs all of its surveillance. Whenever the RIC completes a Mode S or ATCRBS interrogation cycle, the RIC is commanded to squitter listen. Prior to issuing requests for Mode S or ATCRBS interrogations to the RIC, the RIC is requested to stop squitter listening. The RIC then interrupts the Z8002 to signal that it is waiting for a new command. The Z8002 copies the squitters received since the last request from the RIC and issues the new request. The squitters copied from the RIC are added to a list to be transferred to the air-carrier TCAS.

3.2.6 TEU Data Communications

To expedite the completion of GATCAS design, the unit was developed using the TEU software to process target replies rather than translating existing software into the GATCAS unit. The GATCAS is interfaced to the TEU with an RS232 9600 baud link. This link is used to evaluate the GATCAS unit using Version 8 software modified for this purpose. Connection to the TEU is through the TOD (time-of-day) clock port using the original TOD interface card. TOD software was removed and new input/output handlers installed. The following sections describe the communications interface between the GATCAS and TEU.

3.2.6.1 Interface Protocol and Formats

The data link between the TEU and the GATCAS is structured so that the TEU initiates all transfers by sending a request to the GATCAS. The GATCAS responds by sending a reply to each requests. The TEU waits for a reply before sending another request.

There are four possible requests that the TEU can send. They are the ATCRBS request using the top antenna, ATCRBS request using the bottom antenna, squitter request, and Mode S request. The ATCRBS requests cause an ATCRBS interrogation to be sent and the replies sent back to the TEU. The squitter request causes a single squitter, if any, to be sent back to the TEU. Squitter requests are repeated until all squitters have been transferred to the TEU. The Mode S request causes a Mode S interrogation followed by a transfer of a reply, if any, to the TEU.

A request is composed of a header together with an optional data block. The header contains 4 bytes and the data block, when present contains 14 bytes for Mode S requests and 6 bytes for ATCRBS requests. The 6 bytes used for ATCRBS requests have the same format as the first 6 bytes of the Mode S data block. Squitter requests do not have data blocks. Figure 3.2-2 shows the byte assignments for the request formats.

The format of the reply sent in response to each request is similar to the request format. Each reply is composed of a 4 byte header and an optional data block. The header is sent alone whenever there is not a reply to return in response to a request. The data block returned in reply to Mode S or squitter requests is fixed and contains 18 bytes. Six bytes are returned to the TEU for each ATCRBS target responding to an ATCRBS request. Since the number of ATCRBS targets replying to an ATCRBS request is variable, the length of the data block returned in response to an ATCRBS request is dynamic. The length of each data block is returned in the 2nd byte of the reply header. The pertinent byte definitions for the reply format are shown in Fig. 3.2-3.

3.2.6.2 TEU Input and Output Handlers

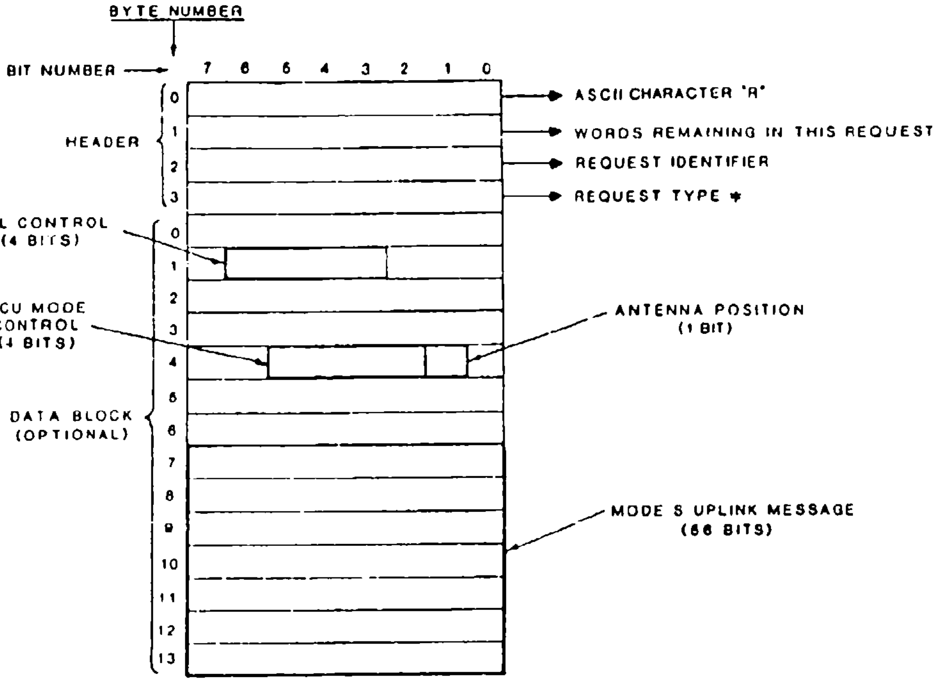
The output handler replaces code in the TEU that sends an MCU command block to the MCU. The handler sends the same MCU command block to the GA unit appended with a 4-byte protocol.

The input handler receives reply data from the GA unit, stores it in the appropriate reply buffer, and then wakes the task responsible for the data. The input handler is initialized by the output task before every output transfer.

The TEU will be in 1 of 3 states: idle, sending or receiving. If idle and it is time to interrogate, the output handler is awakened and the state set to sending. When the last byte of output has been sent, the state will transition to receiving, enabling the input handler and disabling the output handler. When the input handler receives the last byte of input, the state will be set to idle and the appropriate task awakened. The translation to idle will be forced at start of scan time, allowing the sequence to repeat and a disconnected GA unit to be tolerated.

3.2.6.3 GATCAS Input and Output Tasks

The tasks that perform input and output in the GATCAS unit are the REQUEST task and the OUTPUT task. After the system is automatically configured to receive requests from the TEU, the REQUEST task expects input from the air-carrier TCAS and issues a WAITF for the input buffer to fill with request data. If the input buffer does not fill after a specified amount of time, the input message is ignored. This ensures that any messages that are interrupted and not completed will be rejected.



* 1 INDICATES A MODE S REQUEST

2 INDICATES A SQUITTER REQUEST

3 INDICATES A ATCRBS REQUEST

Fig. 3.2-2. Byte definitions for request formats.

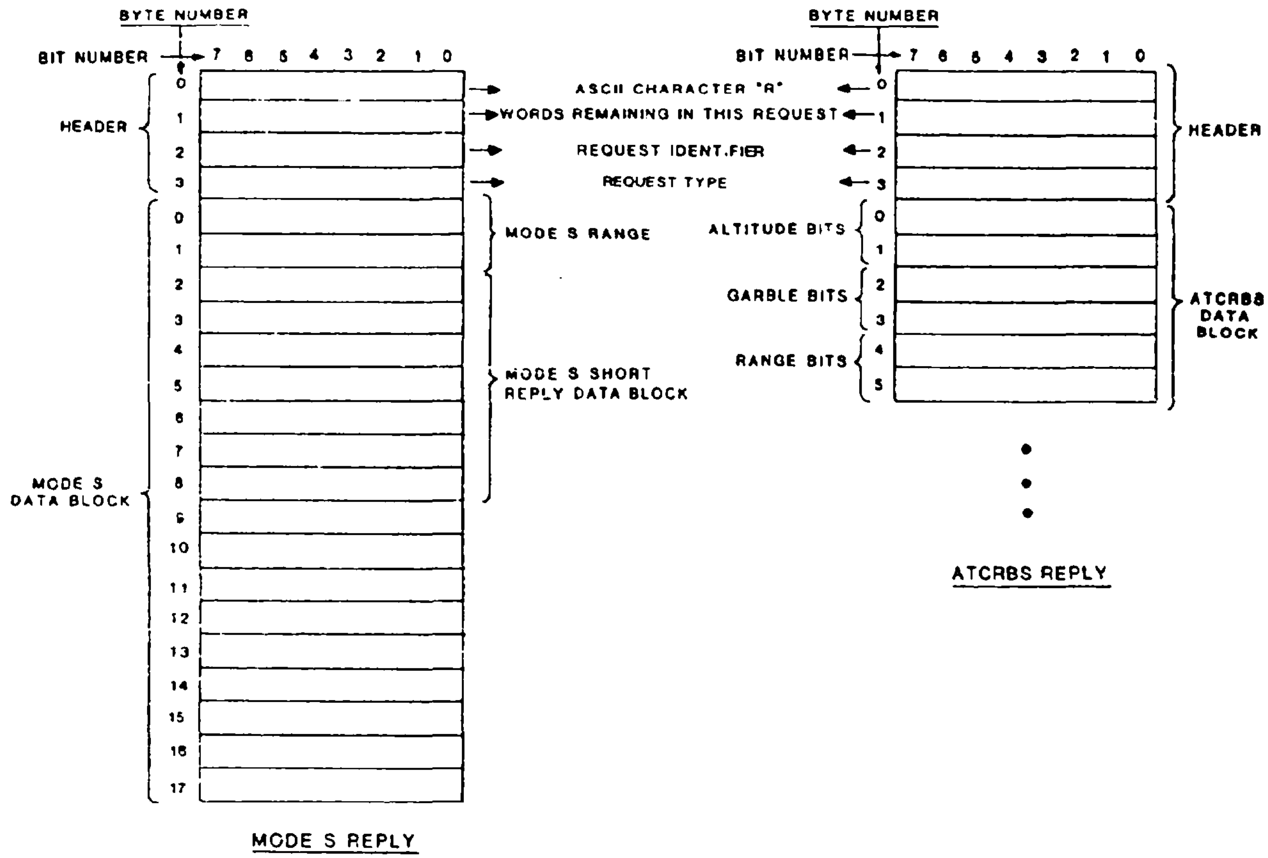


Fig. 3.2-3. Byte definitions for reply formats.

If a request is received for a squitter, the squitter flag is set, the OUTPUT task unblocked, and the first squitter in the squitter list sent to the TEU by the OUTPUT task. When an ATCRBS or Mode S request is received, the appropriate interrogations are sent and the OUTPUT task sends the received replies back to the TEU.

3.3 Real-Time Microprogram Software

3.3.1 General Description

The microprogram software (Fig. 3.3-1) can be divided into four functional parts:

1. ATCRBS interrogation and reply processing
2. Mode S interrogation and reply processing
3. squitter processing
4. diagnostic testing

The first three parts are used for real-time operations while the fourth serves a non-real-time function to test system functionality and integrity. The fourth function is discussed under diagnostic software.

The Am2910 microprogram controller operates as a slave to the Z8002 microprocessor. The Z8002 stores a control word containing the information about a particular task to be done in the control register and then signals the Am2910 to begin processing. The Am2910 reads the control register and performs the required task. When the Am2910 has completed the task, it interrupts the Z8002 and waits for the next request.

3.3.2 Mode C Processing

Figure 3.3-2 shows a flow diagram of the ATCRBS processing routine. When a 0 or 1 is placed in the task control field of the RIC control register and the RIC is requested to begin the current request, the RIC sends an ATCRBS interrogation and gathers replies. A 0 in the task control field causes a Mode C interrogation with no P4 pulse and a 1 in the task control field causes a Mode C interrogation with a P4 pulse. The transmitter is pulse amplitude modulated directly from the RIC pipeline register. Pulse spacing and duration are generated in the microprogram software.

After the interrogations are sent and the transponder turn-around delay is accounted for, the reply detector and Mode C reply accumulator (CRA) are enabled until the CRA memory is full. When the memory is full, the Z8002 is interrupted and the RIC returns to a wait state, waiting for the next request.

3.3.3 Mode S Processing

Figures 3.3-3 and 3.3-4 show the flow diagram for the Mode S mode processing routines. When a 2 or 3 is placed in the task control field (TCF) of the RIC control register and the RIC is requested to begin the current request, the RIC sends a Mode S interrogation and listens for Mode S replies. A 2 in the TCF causes a short Mode S interrogation while a 3 causes a long Mode S interrogation.

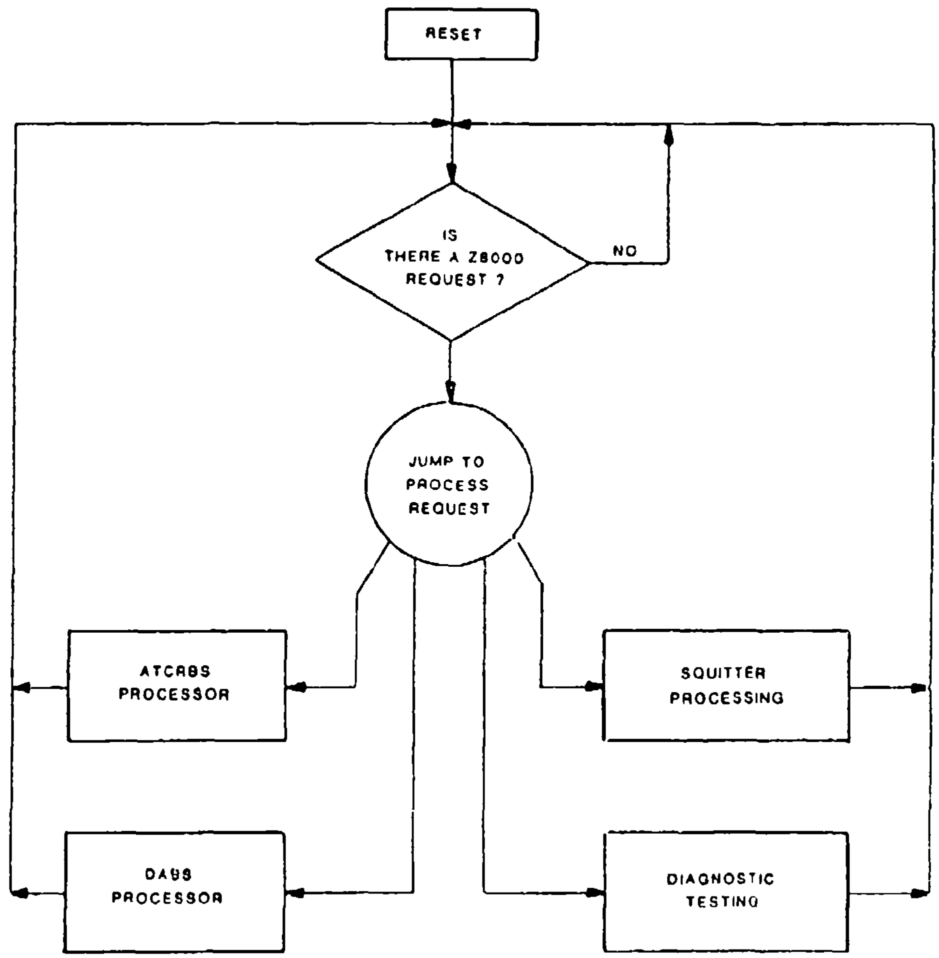


Fig. 3.3-1. Program flow of the AM2910 microprogram software.

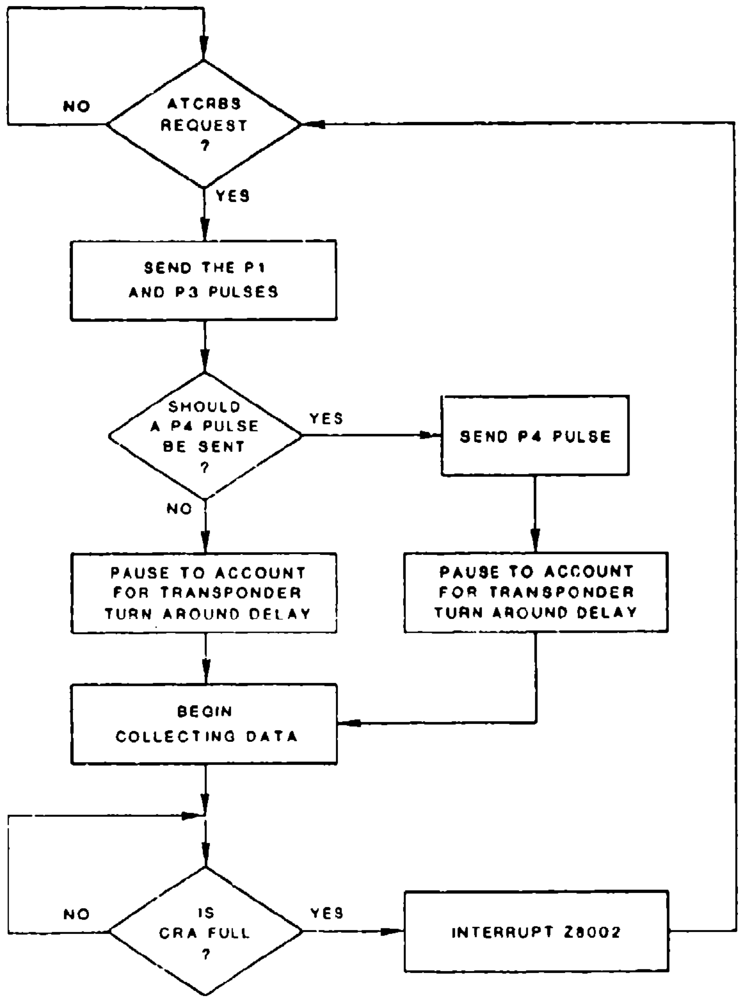
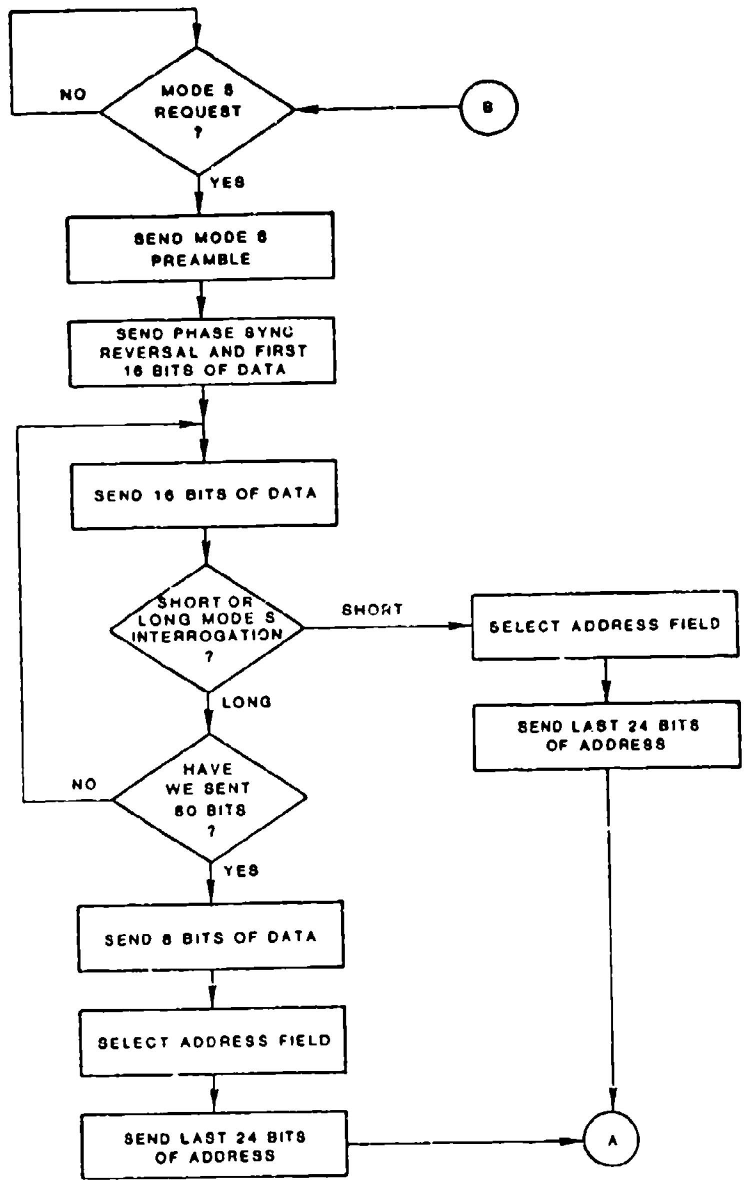


Fig. 3.3-2. ATCRBS processing routine.



ATC-115

Fig. 3.3-3a. Mode S processing routine.

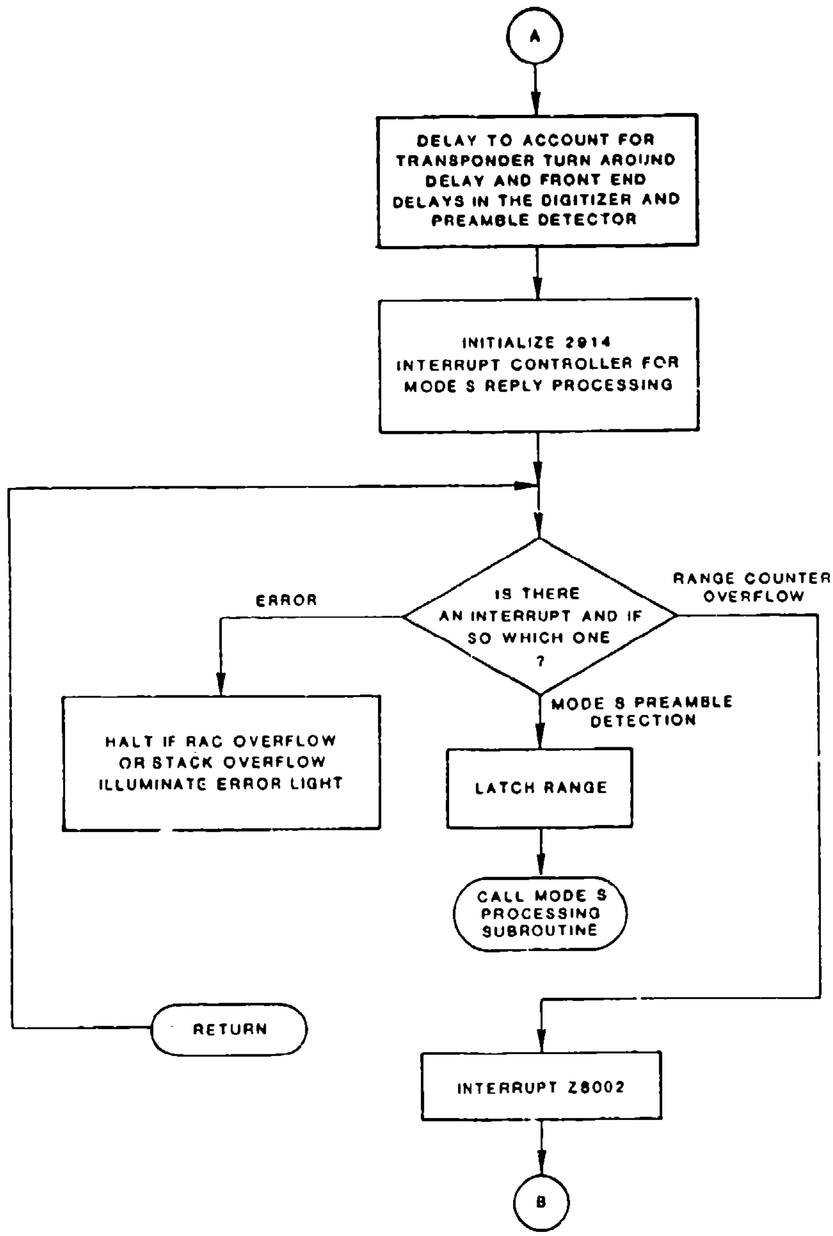
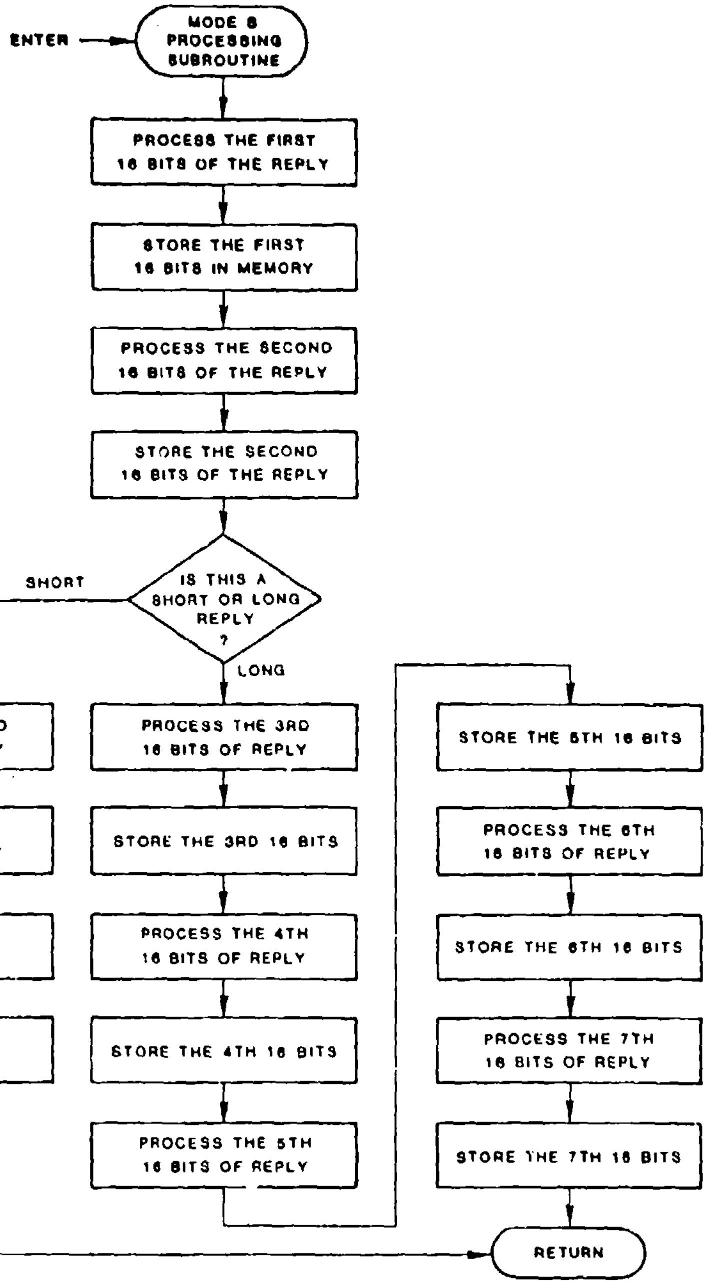


Fig. 3.3-3b. Mode S processing routine.

ATC-115



ATC-115

Fig. 3.3-4. Mode S reply processing subroutine.

The transmitter is again amplitude modulated directly from the RIC pipeline register. The duration and pulse spacing of P1, P2 and P6 are generated in software. The data that drives the transmitter and performs DPSK (differential phase shift keying) modulation during P6 is generated by the Mode S interrogation generator (section 2.6).

As shown in Fig. 3.3-3, when a short Mode S is selected, 32 bits are sent from the Mode S interrogation generator. Following this, the address field is selected and the last 24 bits of encoded address are sent (section 2.6).

If a long Mode S interrogation is selected, 88 bits of data are sent directly, followed by the encoded 24 bits of address. Following either the long or short Mode S interrogation, a delay to account for transponder turn-around and front-end delays in the digitizer and preamble detector is accomplished in software before listening for Mode S replies.

To listen for Mode S replys, the interrupt controller is initialized to listen for one of four separate interrupts. The two highest priority interrupts signal two error conditions, range counter overflow and stack overflow in the Am2910. The action taken in both cases is to halt and activate an error light.

The 3rd highest priority interrupt signals the detection of a Mode S preamble detection. The action taken is to latch the range of the target and call a subroutine to process the Mode S reply. The flow diagram for the subroutine is shown in figure 3.3.4 and is also used for squitter processing.

The lowest priority interrupt is to signal that the range counter has overflowed and Mode S listening can halt. The Z8002 is interrupted and Am2910 returns to a wait state for the next request.

3.3.4 Squitter Processing

Figure 3.3-5 shows a flow diagram of the squitter mode processing routines. When a 4 is placed in the task control field (TCF), the RIC enters the squitter mode when commanded to begin the present request. The RIC remains in this mode until it receives the stop-squittering-command from the Z8002.

When the RIC enters the squitter mode, it sets a flag in the status register so the Z8002 can determine if the RIC is in the squitter mode. While in the squitter mode, the RIC senses three different interrupts marking the occurance of a Mode S preamble detection, a stack overflow in the Am2910, and a stop squittering command from the Z8002. If a stack overflow occurs, the system halts and the error light is illuminated. If a Mode S preamble is detected, a Mode S reply is processed and stored in memory using the Mode S reply processing subroutine. When the stop squitter command is received, the RIC returns to a wait state for the next request.

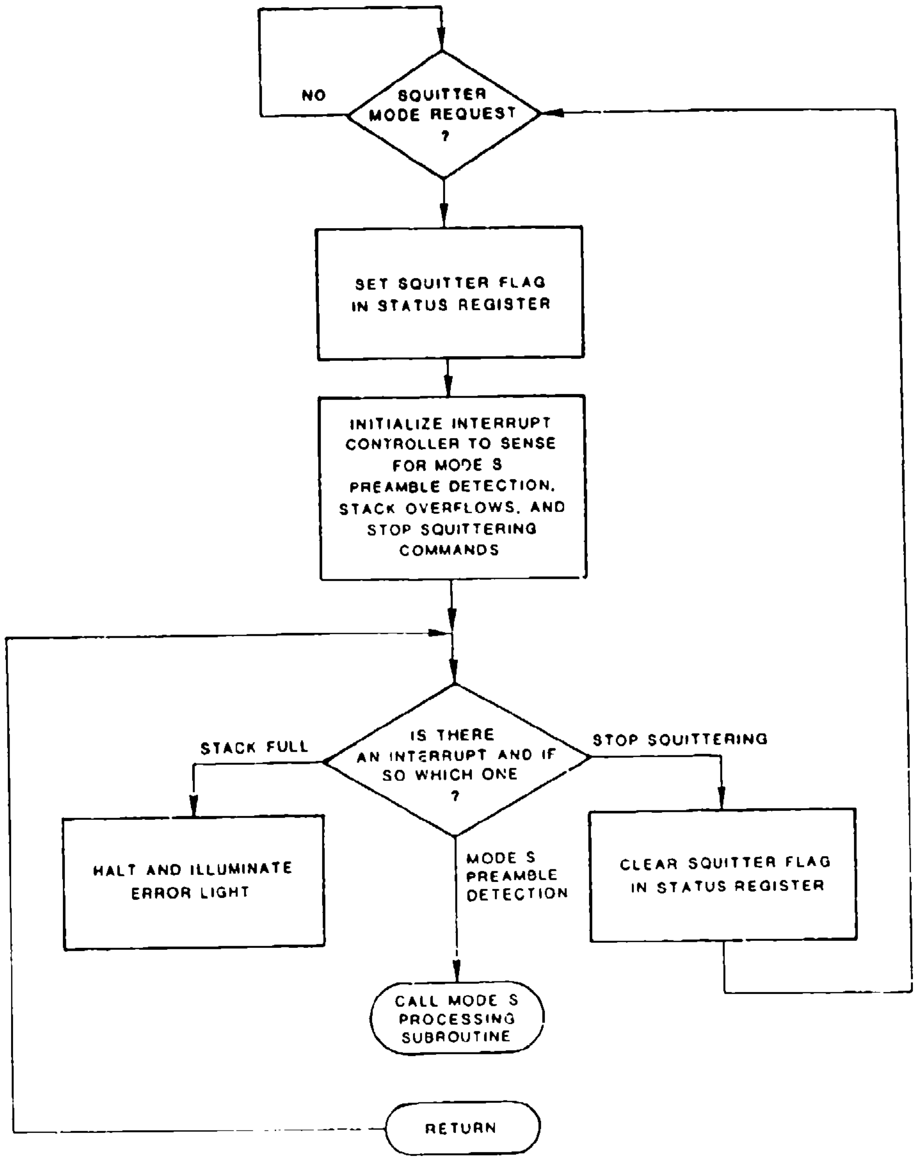


Fig. 3.3-5. Squitter mode processing routine.

4.0 DIAGNOSTIC HARDWARE

To allow the GATCAS unit to test itself for a malfunction, the Mode C reply accumulator (CRA), Mode S interrogation generator (DIG), and Mode S reply processor (DRP) were designed to allow diagnostic tests to be run by the RIC and verified by the Z8002. The hardware design allows a message stored in the RIC data memory starting at location 256 to be encoded, and sent to the Mode S reply processor. The Mode S reply processor then decodes the message and stores it in the RIC data memory at location 0. The Z8002 can then compare the message sent and received to validate the uplink encoder and downlink decoder operation. The DIG and DRP are also designed to allow a complete memory image of the Mode S interrogation area (predefined by the Z8002) to be copied into the Mode S reply area by the RIC. The Z8002 can then read the Mode S reply area and validate Mode S data memory integrity.

The CRA has been designed so that the RIC can fill the CRA memory with all ones or zeroes. The Z8002 can then read the CRA memory to validate CRA memory integrity.

The computer subsystem contains 31 kilobytes of RAM and 160 bytes of PROM. Upon reset, the Z8002 begins executing a program in PROM and tests the RAM for validity. It there are no faults in the RAM, the real-time system initializes and begins running. The software that makes use of the internal hardware diagnostic features is described in Section 5.0.

In addition to internal tests, the unit can be tested for system functionality by directly connecting a Mode S transponder directly to the unit through a radio frequency attenuator (see Fig. 4.0-1). The system can then be operated normally with a real target at zero range. The software designed to exercise the system while it is connected to the transponder is described in Section 5.0.

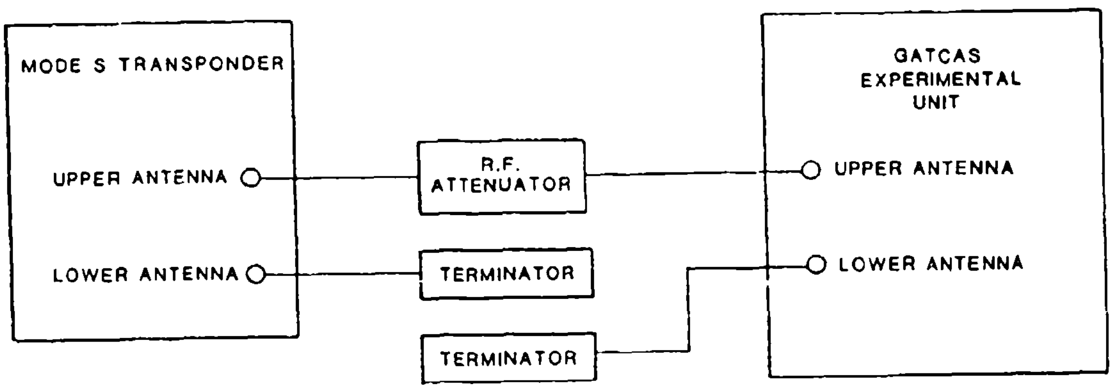


Fig. 4.0-1. Laboratory test configuration for GATCAS unit.

5.0 DIAGNOSTIC SOFTWARE

5.1 General Description

The diagnostic software used to monitor and test the GATCAS operation can be divided into three groups:

1. Z8002 system diagnostics: routines used when both the TEU and the signal processor are logically connected to the Z8002 computer subsystem.

2. RIC diagnostics: routines used when only the signal processor is logically connected to the Z8002 computer subsystem.

3. TEU experimental unit diagnostics: routines used when only the TEU is logically connected to the Z8002 computer subsystem.

On reset, the Z8002 monitor program runs a memory check on the RAM in the computer subsystem. If it fails, and a console device is connected, an error message is printed. If there is no error, the INIT task initializes the other tasks to process ATCRBS, Mode S, or squitter requests from the TEU and starts running in a normal real-time mode. The operator at this time can use the commands described in section 5.2 (Z8002 system diagnostics) to check system operation. If further checks on individual parts of the system are required, normal operation can be terminated and the unit placed in a local mode. In the local mode, the RIC or the TEU can be logically connected and diagnostics run to test each (see sections 5.3 and 5.4).

5.2 Z8002 System Diagnostics

The system diagnostics provided are used to monitor the value of certain state variables and display "snapshots" of the Mode S, ATCRBS, and squitter messages going to and from the TEU. The use of the commands described below requires that both the RIC and TEU unit be logically connected, which is the system state following a reset.

Typing CQ (capital letters required) followed by a carriage return causes the system to type the display:

```
WTB1 #### WTB2 #### REQUEST PC: #### FLG: SSSSSS
INT: DBS=#### SQT=#### ATC=#### BSY=####
QRIC: ## QROLM: ## QXROLM: ## QBANT: ##
CTL: #### UPL: ################################
DNL: ################################
#### RQ=#### D=#### #### S=#### A=#### #### E=####
REPCTR: #### RNGCTR: #### STATUS A: ## D: ## G: ## B: ##
```

The # symbol represents a hexadecimal number and the S symbol represents an alphameric character.

The WTB fields sum together to equal the total number of (hexadecimal) RIC and Z8002 simultaneous memory accesses to signal processor RAM. This is not an error but an indication that memory access conflicts are occurring. The REQUEST PC field gives the hexadecimal address at which the request task is waiting and the FLG field gives the alphameric name of the flag for which the request task is waiting.

The second line gives the status of the interrupts coming into the Z8002. The DBS, SQT, ATC, and BSY fields respectively give the hexadecimal numbers of Mode S, SQUITTER, ATCRBS, and RIC interrupts that have been received since the last reset.

The third line indicates the state of the logical connection between the RIC, Z8002, and TEU. A minus one (hexadecimal FF) in a parameter field indicates that a particular part of the system is logically connected. When a minus one is in the QRIC or QROLM field, the RIC or air-carrier TCAS unit is logically connected. When a minus one is in the QXROLM, the air-carrier TCAS is connected to receive data only. When a minus one is in the QBANT field, the system will use the bottom antenna only.

The fourth and fifth lines contain the CTL, UPL and DNL fields. The CTL field contains the hexadecimal value of the control register. The UPL and DNL fields display the data last sent on the Mode S uplink and received on the Mode S downlink.

The sixth line begins with the number of messages that have been sent to the TEU. The RQ field gives the total number of request that have been received from the TEU. The D, S, and A fields contain the number of Mode S, SQUITTER, and ATCRBS requests respectively, that have been received from the TEU. The E field contains the number of unidentifiable requests that have been received from the TEU due to errors.

The last line contains the Mode S reply address counter (REPCTR), the range counter (RNGCTR), and the STATUS register value. The STATUS field is recorded following a ATCRBS, Mode S, SQUITTER, or BUSY interrupt. The current value is recorded in the A, D, S, or B fields respectively.

The operator can also display the messages that have passed between the GATCAS and TEU during the last few seconds by typing CPK1, CPK2, or CPK3. The CPK1 command collects Mode S messages and then prints them out. CPK2 will do the same as CPK1 but includes squitter messages. CPK3 causes Mode S, squitter, and ATCRBS messages to be displayed. In front of messages going to the TEU unit a < symbol is printed and in front of messages coming from TEU messages a > is printed.

5.3 RIC Diagnostics

To test the operation of the RIC, the unit can be placed in the local mode by typing CRO. This logically disconnects the TEU and allows individual testing of the ATCRBS Mode and Mode S. The CRO is a toggle action command so that the next CRO typed will reconnect the TEU.

In the local mode, the CQ command is still functional and can be used to monitor interrupts from the RIC. If an interrupt from the RIC is not received in response to an RIC request, the error message RIC TIMED OUT is displayed on the console. The CQ command can then be used to determine what interrupt didn't arrive. This should occur only in the local mode. If the RIC does time out and it is desirable to continue, the CID, CIS, CIA, or CIB command can be used to artificially generate the Mode S, squitter, ATCRBS, or BUSY interrupts, respectively.

The commands CMTO, CMTI, . . . CMKTF can be typed in the local mode to set the MTL level to a hex value of 0, 1, . . . F, respectively. The MTL value of 0 corresponds to a receiver sensitivity of -72 dBm, while an MTL value of F corresponds to a receiver sensitivity of -40 dBm.

In the local mode, it is possible to select the top or bottom antenna when the system is reset, the system is initialized to the top antenna. CBA typed once selects the bottom antenna and CBA typed again reselects the top antenna. This can be repeated as desired.

During RIC testing, it may be desirable to inhibit RIC interrupt to the Z8000. This can be done by CIZD, CIZS, CIZA, or CIZB to inhibit the Mode S, squitter, ATCRBS, or BUSY interrupts selectively.

5.3.1 Mode C Diagnostics

Two types of ATCRBS diagnostics can be run with the GATCAS unit; tests that verify internal functionality and tests that make use of an externally-connected Mode S transponder. The internal tests are RAT1 and RAT2. RAT1 commands the RIC to return all zeroes in the Mode C Reply Accummulator (CRA) and RAT2 commands the RIC to return all ones in the CRA. The Z8002 then reads the CRA to verify if the result from the current command is correct. The CQ command will type out OK or BAD along with the system state.

With the transponder connected, three different tests can be run. The first test sends a single ATCRBS interrogation to the transponder and processes the reply. The reply is displayed by printing the left bracket position in decimal, the unscrambled data bits including bracket pulses in hexadecimal, and the corresponding garble bits. This test is executed by typing [illegible] interrogation with P4 and RAN for interrogations with no P4.

The second test causes the first test to be executed repeatedly with the results being printed following each interrogation. To execute the second test, CTR is typed (to get the test repeat mode) followed by TRA for ATCRBS interrogation with P4 or TRAN for ATCRBS interrogations with no P4. CTR is a toggle type command so that the test can be halted by typing CTR again.

The third test is similar to the second test but differs in that it does not display any information. Interrogations are sent to the transponder approximately every 20 milliseconds and the replies are processed normally. This mode is valuable when trying to observe hardware operations on an oscilloscope.

This test is activated by typing CAR followed by RA for interrogations with P4 and RAN for interrogations with no P4. In the fast repetition mode, RIC interrupts are counted but otherwise ignored since the role of the Z8002 is merely that of a signal source to debug the RIC.

5.3.2 Mode S Diagnostics

Two types of Mode S diagnostics can be run with the GATCAS unit; tests that verify internal functionality and tests that make use of an externally connected transponder. The internal test commands are RDT1 and RDT2. RDT1 is used to validate the integrity of the Mode S data memory. The Z8002 must first fill the interrogation part of the Mode S memory with known data. The RDT1 command then copies the interrogation part (256 bytes) of the Mode S memory into the reply part (256 bytes) and interrupts the Z8002. The Z8002 can then compare the reply part of the memory with the data originally placed in the interrogation section to valid the memory.

RDT2 is used to validate the uplink encoder and the downlink decoder. To use the RDT2 command, the Z8002 first loads a valid uplink message into the interrogation part of the memory. The RDT2 command then causes the uplink message to be encoded, channeled through the downlink decoder, and stored in the reply part of the memory. The downlink message should be identical to the uplink message for valid operation. Following both RDT1 and RDT2, the result of the test can be determined by typing CQ.

With the transponder connected, three different Mode S tests can be run. The first test sends a single Mode S interrogation to the transponder and processes the reply. The transponder reply will be printed if a reply is received and "NO MODE S REPLY" will be printed if a reply is not received. The transponder address must match the address interrogated or a reply will not be received. The default Mode S address which is used after system start-up is DAB505. The Mode S address can be changed using the CNU command. To change the address to 432234, for example, the command line would be CNU432234.

The second Mode S test causes the first test to be executed repeatedly with the results being printed following each interrogation. To excute the second test, CTR is type followed by TRD. CTR is a toggle command so that the test can be halted by typing CTR again.

The third test is similar to the second test but differs in that it does not display any information. Interrogations are sent to the transponder approximately every 20 milliseconds and the replies are processed normally. This test is activated by typing CDR followed by RD. Repeating CDR turns the third test off. It is possible to display distinct Mode S replies and the number of Mode S replies that have been received by typing CDP.

5.4 TEU Diagnostics

To test the operation of the TEU connection, the RIC can be logically disconnected and the unit placed in the local mode by typing CRI. The TEU can then interrogate the GATCAS unit and receive dummy ATCRBS targets, Mode S targets, and squitters.

The ATCRBS targets can be initialized to be at any postion in the Mode C Reply Accumulator (CRA) by first typing TA, which will produce the following prompt:

T {Z/G/START BRKT} (T...)

The operator can then enter T followed by a decimal number to locate an ATCRBS left bracket at the particular range. For example, T529 would locate an ATCRBS reply at address 529 in the ARA. The system then prompts the operator for data bits in hex which are to go into the ATCRBS reply. All previous ATCRBS information can be cleared by typing T2, and initialization can be terminated by typing TG.

Squitters can be initialized by first typing TS, which will produce the following prompt:

Mode SID(T....)

The operator can then enter T followed by the hex address of the squitter. For example, TDAB937 would initialize a squitter in memory with address DA3937. The squitter initialization is terminated by typing TG.

The Mode S targets can be initialized by typing CNU followed by the hex address of the Mode S target. For example, CNU DAB515 would initialize a Mode S target with an address of DAB515.

6.0 FLIGHT TEST RESULTS

The GATCAS unit was flight tested in a Cessna 421 which already contained the TEU unit. Figure 2.1-2 shows the equipment in its flight test configuration. To validate Mode S operation, head-on encounters were flown between the Cessna 421 and a Beechcraft Bonanza equipped with a Mode S transponder. Data from these encounters are discussed in Section 6.1

Targets of opportunity in the vicinity of Teterboro, NJ were used to validate the ATCRBS mode. The ATCRBS data was analyzed and the results are presented in Section 6.2

6.1 Mode S Performance

During the head-on encounters flown between the TCAS-equipped Cessna 421 and a Mode S-equipped Beechcraft Bonanza, the Cessna 421 maintained 3,500 feet and the Bonanza maintained 3,000 feet. The aircraft began the encounters from points 10 nautical miles apart and then flew directly toward each other, crossed over, and then continued out-bound until again reaching a separation of 10 nautical miles. The cross symbol in Fig. 6.1-1. designates the range and altitude track of the Bonanza with the altitude track of the Cessna 421 shown by the dashed lines. The range and altitude of ATCRBS targets in the area at the time are indicated by small dots on the figure. The maximum range plotted is 8.7 nautical miles, the maximum range processed by the GATCAS unit. The Mode S replies are of high quality with missing replies only on the in-bound leg. Similar performance has been observed when using the Bonanza as a target aircraft during TEU testing.

The tracks generated from this data by the collision avoidance system (CAS) logic are shown in Fig. 6.1-2. The CAS tracker is able to track through the reply dropouts resulting in a perfect track. Extensive flight tests to perform a statistical study of the GATCAS performance were not conducted because of the correlation of the encounters with the TEU data already collected, and the lack of a sufficiently large population of Mode S targets.

6.2 Mode C Mode Performance

To validate ATCRBS mode performance, two Cessna 421 flights were conducted to Teterboro, N.J., to collect data on targets-of-opportunity. A total of 2 hours and 27 minutes of data were collected and analyzed. This included both enroute and terminal operations. These data were then analyzed to determine system performance by studying individual cases and compiling statistics characterizing the entire data base.

6.2.1 Detection at Long Range

The GATCAS system is designed to work reliably at a range of 3.4 nautical miles for encounters with closing speeds of less than or equal to 300 kts. The data collected included a number of chance encounters which afford an opportunity to assess the range performance. For example, the encounter with the highest closing rate recorded is shown in Fig. 6.2-1. The closing speed is 480 kts with a point-of-closest-approach (PCA) of 0.5 nmi. The track was

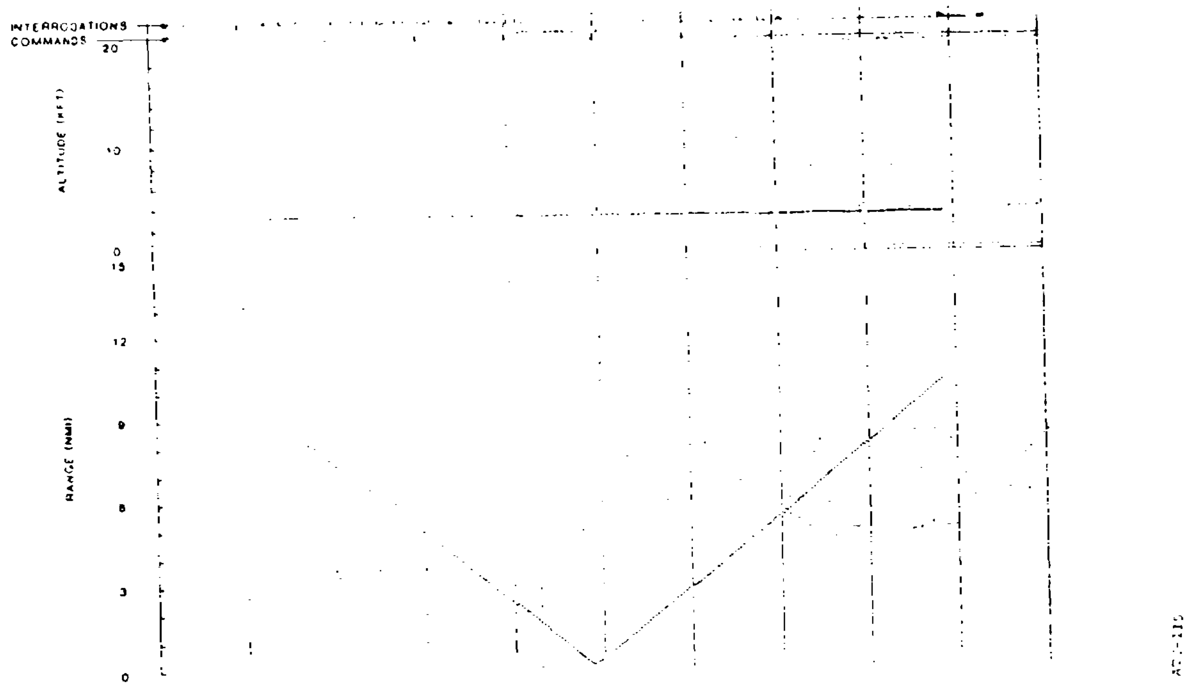


Fig. 6.1-1. Reply data from mode S head-on encounter flight.

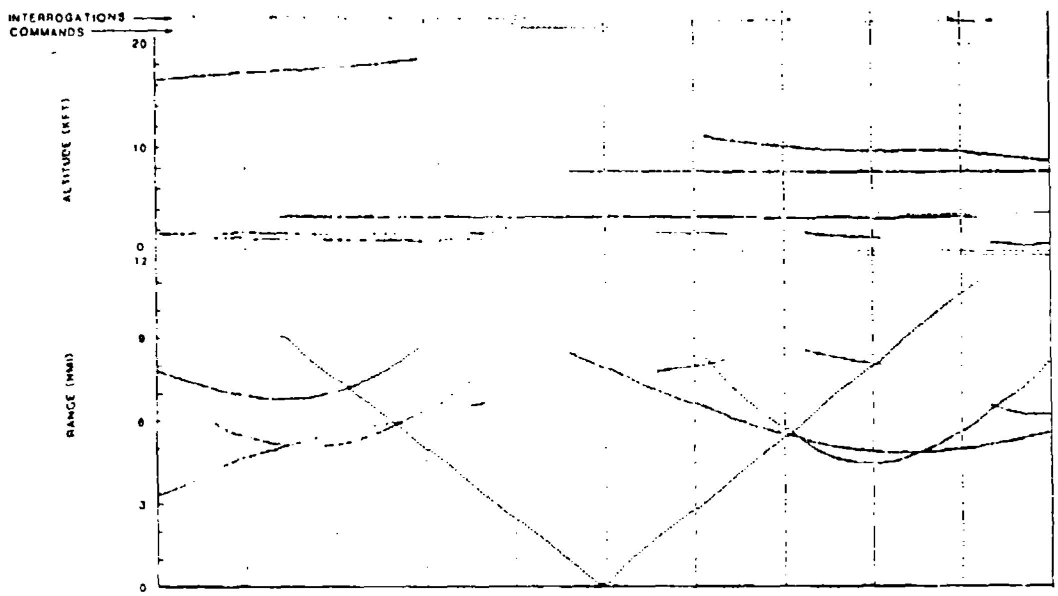


Fig. 6.1-2. CAS tracks from Mode S head-on encounter.

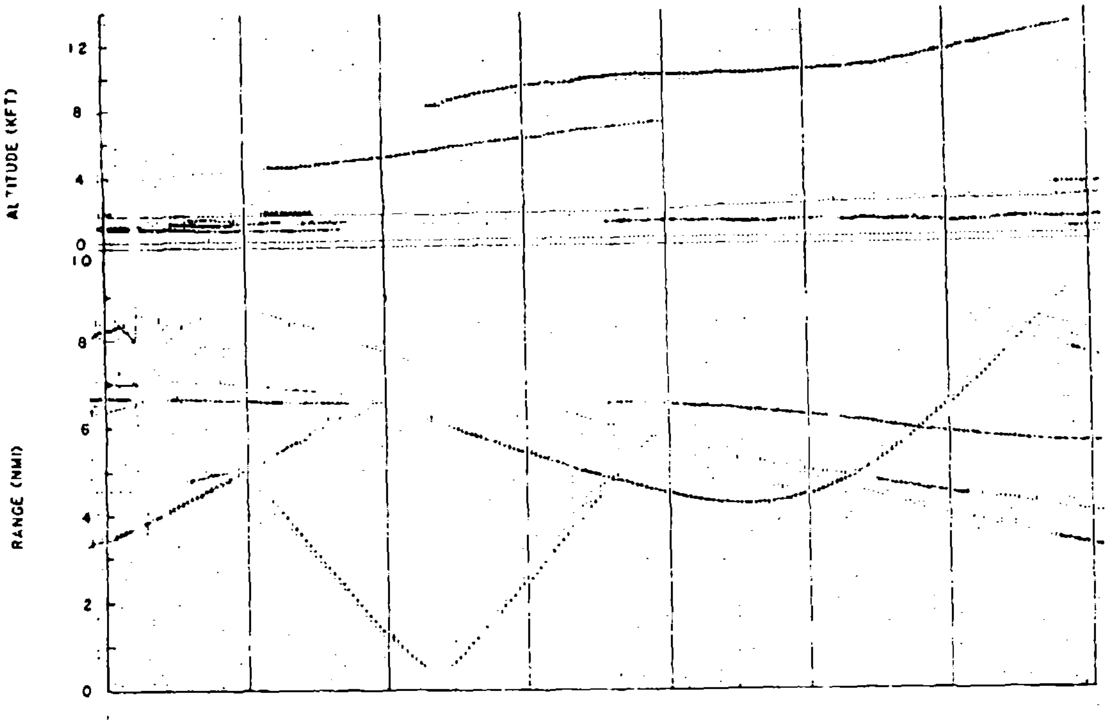


Fig. 6-2-1. CAS tracks from 480-Knot ATCRBS encounter

established at a range of 4.5 nmi, which occurred 35 seconds before the PCA. During the 35 seconds prior to PCA, there were 4 coasts (not shown in the figure). The blip/scan ratio was 88% during this time.

There were a number of other chance encounters with varying closing speeds in the data collected (although due to differences in altitude, many of these encounters did not result in maneuver advisories). Their characteristics with performance measures are listed in Table 6-1. The traffic density present during each encounter, also listed in the table, was calculated by counting the aircraft (other than the subject aircraft) within 8.7 nmi, averaging this count over a one-minute period prior to PCA, and then dividing the result by times 8.7 nmi squared.

The performance observed in the data collected was very good. In all cases, including encounters with closing speeds exceeding the system specification, track was established 35 seconds or more before PCA and the tracks were continued without drop through PCA.

6.2.2 Statistical Performance Assessment

In addition to the individual cases described above, a statistical performance analysis was also conducted on the 2 hours and 27 minutes of data collected during the flights to Teterboro, N.J. The analysis included the determination of the probability of track (POT), probability of report (POR), probability of coast (POC), and a study of performance vs aircraft density.

6.2.2.1 Performance Definitions

The performance measures used in this report are defined as follows:

Probability of track: For a given scan and a particular aircraft of interest, the probability that an established track of that aircraft exists on that scan.

Probability of report: For a given scan and a particular track of interest, the probability that the track is updated with a report on that scan.

Probability of coast: One minus the probability of report.

6.2.2.2 Probability of Report

Probability of report was evaluated from the BEU data base by computing the ratio of number of reports to the sum of number of reports and coasts (total scans). The ratio was evaluated as a function of two variables, range and number of overlaps. Range is divided into three intervals, 0-2 nmi, 2-4 nmi, and 4-6 nmi. The number of overlaps is defined as the number of aircraft with ranges within 1.67 nmi of the subject aircraft range. Aircraft further apart in range cannot produce replies that overlap in time. Both altitude-reporting and non-altitude-reporting aircraft were considered when determining overlapping aircraft.

TABLE 6-1.

ATCRBS MODE PERFORMANCE.

Case	TCAS Alt.	Other ALT	Density Aircraft	Closing Speed	PCA	Acquisition Range	Acq. Time	Track Continuity	Coasts Inbound
A	1600 ft.	5000	.017	480	0.5	4.5	35	100%	4/35 = 11%
B	4000	3000	.017	240	0.1	8.4	127	97%	35/127 = 26%
C	3700	4000	.013	300	0.8	8.4	106	100%	20/106 = 19%
D	3700	6000	.006	340	2.0	8.4	99	100%	17/99 = 17%
E	7400	9000	.006	430	1.0	6.0	56	100%	6/56 = 11%
F	5100	27000	.013	200	0.8	5.0	87	100%	53/87 = 60%

Rather than evaluating probability of report over the entire trajectory of every real aircraft, the evaluation was conducted for aircraft within a region of interest. The time an aircraft spent within 600 feet of ground level was not counted, nor was the time it spent outside 10° in elevation angle. Performance when either aircraft was near the ground (less than 500 feet) is excluded from this study simply to focus attention on the primary region in which GATCAS is intended to operate.

The results are shown in Table 6-2. As expected, probability of report degrades with increasing number of overlaps and longer range.

6.2.3 Probability of Track

The most important performance measure is probability of track. The probability of track is determined by dividing the total number of scans a recorded track was maintained on a target by the total number of scans a track should have been maintained. To accomplish this, it would be desirable to have an independent source of surveillance to determine the presence of aircraft. Since an independent source was not available, the only course of action was to apply a superior tracker to the same reply date. This was done manually, using plots of the reply ranges and altitudes versus time. By concentrating primarily on the range plots inside six miles, the existence of aircraft could be confidently inferred even when round reliabilities were as low as about 25%. Gaps as long as tens of seconds having even lower round reliabilities were confidently filled in on the basis of only a few replies.

As in the preceding section this analysis was limited to aircraft in the region of interest. The results of comparing the real aircraft trajectories to the GATCAS tracks are shown in Table 6-3. Performance is seen to be very good in the most important region within 3.4 nmi. As expected, probability of track is best at short ranges while degrading gradually at longer ranges.

6.2.4 Performance as a Function of Aircraft Density

An indication of the aircraft target densities during these flights is given in Fig. 6.2-2, a histogram of the number of targets in track. These figures refer to the number of aircraft within 6 nmi, and include altitude reporting aircraft only. The equivalent target densities marked in the figure are based on the formula:

$$\text{density} = \frac{\text{Number of aircraft exclusive of the TCAS aircraft}}{\pi\,(6\ \text{nmi})}$$

The effect of aircraft density on probability of track for aircraft of interest was evaluated by dividing the range into three intervals; 0-2 nmi, 2-4 nmi, and 4-6 nmi. Tracks within each range interval were examined continuously to determine the local density within the range interval during

TABLE 6-2.

PROBABILITY OF REPORT EVALUATED FOR AIRCRAFT OF INTEREST

	Range		
No. of Overlaps	0-2 nmi	2-4 nmi	4-6 nmi
0	.89	.81	.81
1	.76	.73	.70
2	N/A	.65	.62

TABLE 6-3.

PROBABILITY OF TRACK EVALUATED FOR AIRCRAFT OF INTEREST

	Range		
	0-2 nmi	2-4 nmi	4-6 nmi
Probability of Track	.92	.82	.72

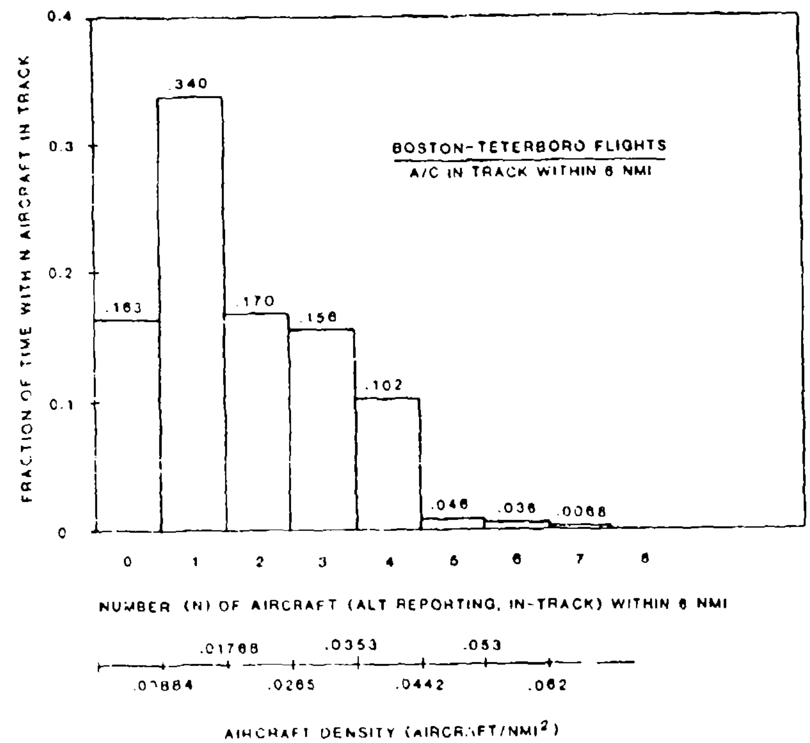


Fig. 6.2-2. Histogram of the number of targets in track.

each scan. Probability of track was then determined by tabulating the aircraft-seconds for which there existed a track during all observed densities. The results are shown in Table 6-4. The last column gives the probability of track over the range 0-6 nmi which is the same result given in Table 6-3.

The data of Table 6-4 is plotted vs interference density in Fig. 6.2-3. The interference density differs from the track density in that the track density excludes only the TCAS aircraft from the aircraft of interest while the interference density excludes both the TCAS aircraft and the target aircraft. The interference density is calculated using the formula:

$$\text{density} = \frac{\left\{\begin{array}{l}\text{Number of aircraft exclusive} \\ \text{of the TCAS aircraft}\end{array}\right\} - 1}{\pi \times (6\ \text{nmi}^2)}$$

From Fig. 6.2-3, it may be observed that the variance of this statistical data is high due to the small amount of data collected. However, the data does illustrate functionality. Performance is good at low density over the range considered, with the performance at shorter ranges being comparable with that of the TEU. Performance at 4-6 nmi range degrades significantly as the target density increases. (The solid curves included in this figure are estimates of the trend of the data points for each range interval). The results indicate that reliable TCAS surveillance was achieved out to a range of 4 nmi for densities less than 0.08 interferers per square nmi. Higher densities or longer ranges would likely require additional degarbling capability and higher transmit power.

TABLE 6-4

PROBABILITY OF TRACK VS. TARGET DENSITY EVALUATED FOR AIRCRAFT OF INTEREST

		Number of ATCRBS Transponders Within 6 nmi Excluding the TCAS Equipped Aircraft					
		1	2	3	4	5	1 through 5
Range Interval (nmi)	0-2	28 314 .92	0 110 1.0	5 17 .77	10 47 .82	N/A	43 485 .92
	2-4	145 793 .85	182 737 .80	33 123 .79	18 81 .82	N/A	378 1734 .82
	4-6	244 1400 .85	446 721 .62	136 278 .67	52 32 .38	57 10 .15	935 2441 .72

Note: The three entries in each case are

(a) number of aircraft-seconds for which there was no track

(b) number of aircraft-seconds for which there was a track

(c) probability of track = $\frac{(b)}{(a)+(b)}$

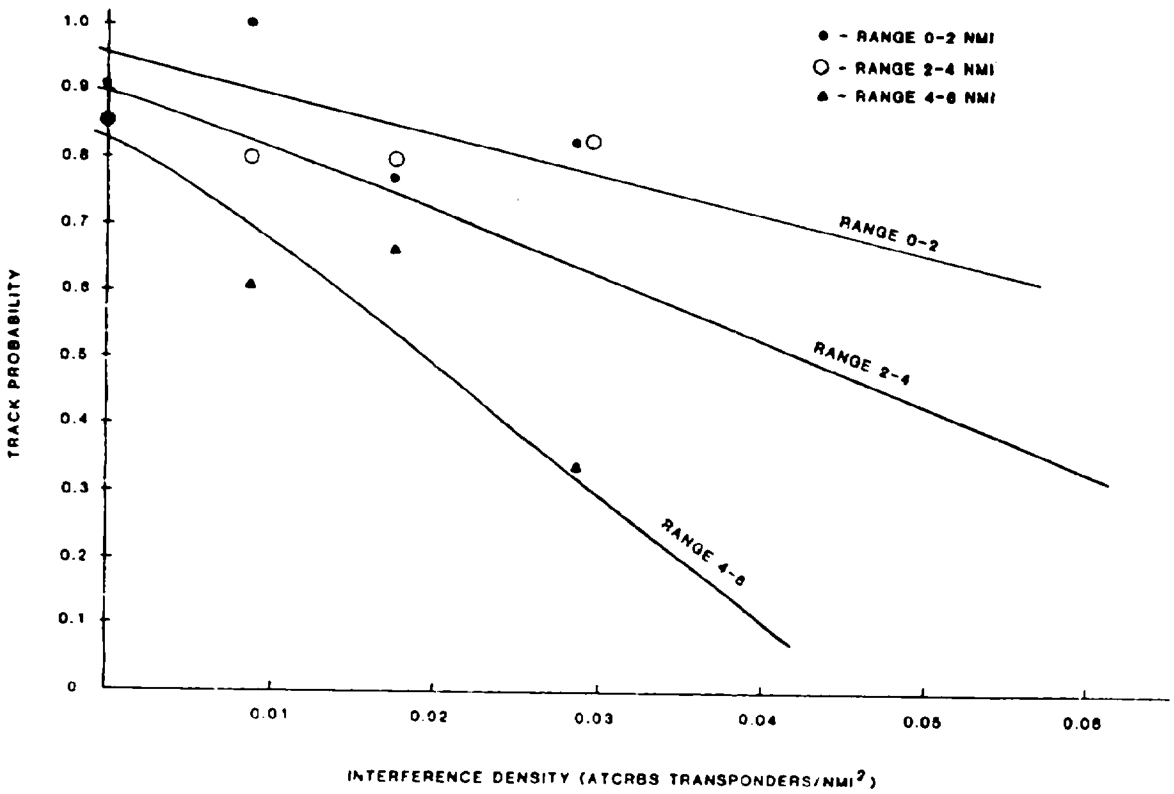


Fig. 6.2-3. Track probability vs interference density.

REFERENCES

1. "Z8000 User's Manual," Advanced Micro Devices Inc. publication number AMZ-291, Copyright 1981.

2. "Am96/4016 AmZ8000 Evaluation Board User's Manual" Advanced Micro Devices Inc. publication number 00680131, Copyright 1979.

3. "MOS/LSI Data Book" Advanced Micro Devices Inc. publication number AM-PUB118, Copyright 1980.

4. R. G. Nelson and J. H. Nuckols, "A Hardware Implementation of the ATCRBS Reply Processor Used in DABS", Project Report ATC-78, Lincoln Laboratory, M.I.T. (September 1977), FAA-RD-77-92.

5. J. L. Gertz, "The ATCRBS Mode of DABS", Project Report ATC-65, Lincoln Laboratory, M.I.T. (31 January 1977), FAA-RD-76-39.

6. "The Am2900 Family Data Book" Advanced Micro Devices Inc. publication number AM-PUB003, Copyright 1979.

7. S. Kowalski, et al, "Cost Analysis of the Discrete Address Beacon System for the Low Performance General Aviation Aircraft Community", DOT/FAA/RD-81/61, Final Report (September 1981).

8. K. Markin, D. Swann, "Cost Development of the Dual-Channel GPS Navigator for General Aviation Application", ARINC Research Corporation Draft Report Covering Work Performed Under FAA Contract DTFA01-80-C-10030 (August 1982).

APPENDIX A

GATCAS COST ESTIMATE

To determine the cost of manufacturing a general aviation TCAS unit similar to that described in this report, a combination of methods was used. The system assembly was divided into seven separate areas. When possible, the costs were estimated based on an itemized list of the parts required.

When this method could not be applied, a cost was derived by comparison with two other types of general aviation avionics previously analyzed by ARINC Research Corporation. The two studies used for comparison were the "Cost Development of the Dual-Channel GPS Navigator for General Aviation Application" [7], and the "Cost Analysis of the Discrete Address Beacon System for the Low Performance General Aviation Aircraft Community" [8].

The seven categories into which the construction of a GATCAS unit were divided are:

1. Transmitter
2. Receiver
3. Video Pulse Quantizer
4. Digital Logic
5. Power Supply
6. Enclosure and Chassis
7. Assembly and Test

Table A-1 lists the cost estimate for each of these areas along with allowances for labor, administrative expenses, and profit. The data for determining material handling costs, labor charges, factory overhead charges, quality control costs, administrative expenses, profit, and distribution were obtained from ref. [7].

The transmitter and receiver costs were based primarily on the costs found in ref. [8] for a DABS transponder with Comm A, B, C, and D. This was necessary because the GATCAS unit did not use RF construction techniques employed in commercial units. The costs found in the ARINC study were increased to account for a more stable microwave oscillator, a pulse amplitude modulation switch, a diversity switch, and an additional low-pass filter.

The Video Pulse Quantizer (VPQ) and the Digital Logic cost estimates were evaluated using the parts-cost method. This was possible because both were designed and built at Lincoln Laboratory and detailed parts lists were available. Detailed bills of material and associated labor units were prepared for each and the material costs were determined based on the largest quantity prices available (generally quantities of 1,000 or greater).

TABLE A-1
GATCAS COST SUMMARY

COST ELEMENT	MODULE COST IN 1982 DOLLARS							
	Transmitter	Receiver	Video Pulse Quantizer	Digital Logic	Power Supply	Enclosure & Chassis	Assembly & Test	Totals
Material Cost	261.35	76.70	565.14	1140.99	50.03	86.37	-	2180.58
Material Handling (10%)	26.14	7.67	56.51	114.10	5.00	8.63	-	218.05
Labor (7.64/Hr)	30.56	26.74	39.13	103.34	22.93	33.09	49.50	305.29
Burden (135%)	41.25	36.10	52.83	139.50	30.96	44.67	66.83	412.14
Subtotal	359.30	147.21	713.61	1497.93	108.92	172.76	116.33	3116.06
Admin. (27%)	97.01	39.75	192.67	404.44	29.40	46.64	31.41	841.32
Total Direct Cost	456.31	186.96	906.28	1902.37	138.32	219.40	147.74	3957.38
Profit (15%)	68.45	28.04	135.94	285.35	20.74	32.91	22.16	593.59
Factory Sell Price	524.76	215.00	1042.22	2187.72	159.06	252.31	169.90	4550.97
Distribution (100%)	-	-	-	-	-	-	-	4550.97
List Price	-	-	-	-	-	-	-	9101.94

The power supply costs for the GATCAS unit were found by adding the costs found in the GPS and transponder studies. Both the material costs and the hours of labor were directly added with no reduction. The cost for labor was then determined using a rate of $7.64/hour. This approach is conservative but valid since the GATCAS RF subsytems are very similar to those of the transponder, and the GATCAS digital logic requirements are nearly identical to those of the GPS unit. The percentage of the overall cost due to the power supply is also very small.

The enclosure and chassis cost was derived similarly to the power supply cost except that the cost for materials and the hours of labor taken from the GPS study were scaled down by a factor of .56. This scale factor was found by taking the ratio of the total printed circuit board areas of the GATCAS and the GPS unit. The labor cost was then calculated using $7.64/hour.

The assembly and test cost for a GATCAS unit was found by adding the hours of labor found in the GPS and transponder studies and then using the rate of $7.64/hour.